ЖУЙКОВ В.Я., ТЕРЕЩЕНКО Т.О., ЯМНЕНКО Ю.С., МОРОЗ А.В.

РЕГУЛЬОВАНІ ФІЛЬТРИ ДЖЕРЕЛ ЖИВЛЕННЯ ДЛЯ ЗАХИСТУ ІНФОРМАЦІЇ В МІКРОКОНТРОЛЕРАХ

Київ – 2016

УДК 621.3.3.037.37

ББК

Рецензенти

Г.Г. Жемеров, д-р техн. наук, професор
(Національний технічний університет «Харківський політехнічний інститут»)

О.М. Юрченко, д-р техн. наук, пров. наук. співробітник
(Інститут Електродинаміки НАН України)

Рекомендовано до друку Вченою Радою Національного технічного університетуУкраїни «Київський політехнічний інститут»
Протокол № від

Жуйков В.Я., Терещенко Т.О., Ямненко Ю.С., Мороз А.В. Регульовані фільтри джерел живлення для захисту інформації в мікроконтролерах. Монографія. – Київ , 2016 – 184 с
ISBN

Описано нові схеми та алгоритми регульованих фільтрів живлення мікроконтролерів із маскуванням інформаційних сигналів, в основу функціонування яких покладено вейвлет та спектральний аналіз в полярних координатах та досліджено їх ефективність.

Книга призначена для науково-педагогічних працівників, студентів, аспірантів та інженерів, що займаються розробкою мікропроцесорних та електронних систем

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

DPA	-	Differential Power Analysis, Диференційний аналіз живлення
DES	-	Data Encryption Standard, Стандарт шифрування даних
DSP	-	Digital Signal Processor, Цифровий сигнальний процесор
HO-DPA	-	High Order Differential Power Analysis, диференційний аналіз струму живлення високого порядку
RSA	-	Rivest, Shamir, Adleman, Рівест, Шамір, Аделман, несиметричний алгоритм шифрування названий за прізвищами винахідників
SPA	-	Simple Power Analysis, простий аналіз живлення
ГВС	-	Генератор випадкових станів
ГВЧ	-	Генератор випадкових чисел
МК	-	Мікроконтролер
МП	-	Мікропроцесор
ОБ	-	Перетворення в орієнтованому базисі
ПЗ	-	Програмне забезпечення
СКІ	-	Перетворення на скінченних інтервалах
ЦП	-	Центральний процесор

ВСТУП

В сучасних електронних пристроях постійно збільшується обсяг обробки інформації, включаючи конфіденційну. Доступ до такої інформації може створити небажані наслідки для власника, наприклад викликати грошові збитки. Через це важливою та актуальною є проблема захисту інформації в мікроконтролерах та мікропроцесорах від несанкціонованого доступу. Одним із можливих способів несанкціонованого отримання інформації є зчитування та аналіз струму споживання. Одержання характеристики струму споживання мікропроцесора при виконанні критичних частин програми може надати зловмиснику додаткову інформацію при несанкціонованому доступі до системи. Нагальною задачею розробки системи з мікроконтролерами є забезпечення необхідного захисту програмного забезпечення та даних, що знаходяться у пам’яті і являє собою об’єкт авторського права або конфіденційну інформацію.

При розробці мікропроцесорних систем керування перетворювачами електроенергії основна доля трудомісткості в більшості випадків припадає на програмне забезпечення, оскільки апаратні частини систем керування є практично стандартними. Для захисту авторських прав розробники використовують мікроконтролери із захистом інформації, наприклад з бітами захисту, що ускладнюють зчитування програмного коду. На сьогодні особливо важливим є правильний вибір мікроконтролерів для захищених систем, а також проведення достатнього обсягу тестувань. Відомими методами несанкціонованого доступу до інформації в мікроконтролері є деструктивні та недеструктивні. Деструктивні методи передбачають відкриття корпуса мікросхеми та модифікацію внутрішніх електричних з’єднань, потребують дорогого обладнання: електронних мікроскопів, лазерів, мікрощупів, тому ці методи рідко використовують для зчитування програмного забезпечення мікроконтролерів. Недеструктивні методи дозволяють отримати інформацію

про внутрішній стан мікроконтролера без відкриття корпусу мікросхем і не потребують дорогого обладнання зчитування інформації. Недеструктивні методи атак включають у себе: дослідження часу виконання програми; атаки з повним перебором ключів доступу; генерування електричних завад з метою виклику збоїв; аналіз побічних каналів витоку інформації; простий та диференційний аналіз струму живлення. Захист від кожного недеструктивного методу атак повинен реалізовуватися з урахуванням особливостей кожного методу.

Найбільший інтерес при несанкціонованому доступі зловмисників до інформації є методи атак за струмом споживання, такі як простий аналіз струму споживання (SPA - Simple Power Analysis) та диференційний аналіз струму споживання (DPA - Differential Power Analysis), оскільки вони легкі для виконання, мають низьку вартість і можуть бути реалізовані за допомогою тільки цифрового осцилографа та комп’ютера.

Захист інформації від зчитування за струмом споживання за допомогою регульованих фільтрів напівпровідникових джерел живлення тільки починає сьогодні застосовуватися.

Дослідження [1] [2] показали, що загрозі несанкціонованого доступу піддаються пристрої, що використовують криптографічні алгоритми з відкритими ключами для шифрування інформації і аутентифікації користувача. Це можуть бути смарт-карти, що застосовуються для зберігання електронних грошей або персональних даних, картки електронного цифрового підпису, картки аутентифікації користувача для дистанційного доступу до корпоративних мереж. Розроблений [3] метод атаки був застосовний для отримання ключів в криптосистемах з відкритим ключем – RSA, алгоритму цифрового підпису Рабіна, системі ідентифікації Фіата-Шаміра і подібних [4]. Існує програмно-апаратний інструментарій для визначення секретних ключів криптографічних пристроїв за струмом споживання [5].

Монографію присвячено вирішенню задачі захисту мікропроцесорних і мікроконтролерних систем від несанкціонованого зчитування за струмом споживання за допомогою спеціальних регульованих фільтрів напівпровідникових джерел живлення.

Монографія складається з п’яти розділів та додатку. *Перший розділ* присвячено методам захисту інформації в мікроконтролерах. Розглянуто основні принципи та критерії захисту, існуючі методи атак, а також забезпечення захисту мікроконтролерів від несанкціонованого доступу шляхом застосування регульованих фільтрів та спеціалізованих систем живлення. *У другому розділі* розглядаються математичні основи дослідження струму споживання, зокрема, методи аналізу дискретних сигналів, інтерполяції даних, вейвлет-перетворення в полярних координатах. Для характеристики струму споживання запропоновано використання інтегрального показника у полярних координатах, що дозволяє спростити процес ідентифікації команд у мікроконтролері. *У третьому розділі* проведено оцінку ступеню захищеності мікроконтролерів від зчитування за струмом споживання. Для дослідження струмів споживання при виконанні різних команд мікроконтролера застосовано методи кореляційного аналізу та програмне забезпечення. Наведено опис експериментальної установки, що використовувалася для відпрацювання розроблених методів. *Четвертий розділ* присвячений побудові різних типів регульованих фільтрів джерел живлення мікропроцесорних систем із маскуванням струму споживання. *П’ятий розділ* містить матеріал щодо експериментального дослідження запропонованих типів регульованих фільтрів та порівняльного аналізу їх ефективності. *У додатку* наведено математичні співвідношення щодо апарату вейвлет-аналізу дискретних функцій в орієнтованому базисі.

Монографія є узагальненням багаторічного досвіду наукової та дослідницької роботи авторів в напрямку розробки алгоритмів та систем мікропроцесорного керування та цифрової обробки сигналів.

Монографія є корисною для науково-педагогічних працівників, студентів, аспірантів та інженерів, що займаються розробкою мікропроцесорних та електронних систем.

Автори будуть вдячні за конструктивні поради, критику, відгуки та побажання щодо матеріалу монографії.

Адреса для листування: petergerya@yahoo.com

Телефони: +38 044 204 8293, +38 044 236 2117

РОЗДІЛ 1

МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЇ В МІКРОКОНТРОЛЕРАХ

1.1. Рівні захисту інформації

Тенденцією сьогодення є покращення зручності життя людини за рахунок перенесення різноманітних сервісів на електронну платформу. В період світової інформатизації основним об’єктом промислової власності стає інтелектуальна власність. Саме тому важливим стає захист програм від несанкціонованого копіювання, тиражування, використання не за призначенням. Часто фінансовий успіх компанії залежить від надійності захисту програмного забезпечення (ПЗ) мікропроцесорних систем від зчитування та несанкціонованого доступу. Наприклад, успіх компаній платного телебачення напряму залежить від надійності захисту абонентських смарт-карт. В сучасних електронних платіжних картках також використовуються смарт-карти з мікроконтролерами для зберігання інформації про рахунок клієнта. Бодай один випадок несанкціонованого доступу до такої інформації може коштувати занадто багато як у фінансовому плані, так і у морально-психологічному аспекті.

Оцінка захищеності від несанкціонованого доступу для мікроконтролера або мікропроцесора являє собою непросте завдання, оскільки така оцінка вимагає врахування багатьох факторів [6]. Основні з них є:

1) тип пакування (корпуса) мікросхеми;
2) топологія кристала мікросхеми;
3) тип та структура внутрішньої пам’яті;
4) наявність інтерфейсів введення–виведення;
5) алгоритми програмування та відлагодження.

Варто врахувовати наявність різноманітних вбудованих механізмів захисту та безпеки таких як, brown-out детектори, захисні Fuse-біти [7], фільтри

живлення, захисний екран [8]. Немає однозначного тесту чи методу, за яким визначається рівень безпечності або захищеності інформації у мікроконтролері або у мікросхемі пам’яті. Лише після спроби виконати той чи інший тип атаки можливо визначити стійкість системи до даного типу атак.

Фірмою IBM розроблена класифікація систем [8] за рівнем їх захищеності та можливості протидії різним типам атак яка складається з шести рівнів, починаючі з нульового рівня, який відповідає системі без захисту, до найвищого рівня, що відповідає системі, яку практично неможливо зламати

Рівні безпеки наступні:

1) *Нульовий рівень.* Система без спеціальних заходів безпеки. Всі складові системи відкриті для доступу та дослідження ззовні. Прикладом такої системи є мікропроцесор або мікроконтролер із зовнішнім постійним запам’ятовуючим пристроєм (ПЗП).
2) *Низький Рівень.* Деякі засоби захисту інформації використовуються, але вони можуть бути відносно просто зламані за допомогою простих засобів, таких як вольтметр та аналоговий осцилограф. Атака вимагає певного часу, але не вимагає дорогого обладнання. Приклад: мікроконтролер з незахищеною внутрішньою пам'яттю, але зі спеціальним захищеним алгоритмом програмування.
3) *Середньо-низький рівень.*Використовуваний захист захищає від більшості простих та дешевих атак. Для зламу потрібні спеціальні знання та відносно недешеве обладнання. Приклади: мікроконтролери, чутливі до перепадів напруги живлення, завад напруги живлення або чутливі до зміни тактової частоти, тощо.
4) *Середній рівень.* Для успішної атаки потрібні спеціальні знання а також спеціалізовані інструменти та прилади. Атака займає багато часу, являє собою довгий та кропіткий процес. Прикладом таких систем є мікроконтролери із захисним покриттям проти опромінення ультрафіолетом; мікросхеми, що містять захисні механізми для

спотворення струму споживання та стійкі до завад по живленню та по тактовим сигналам.

5) *Середньо-високий рівень.* При проектуванні системи використовуються спеціальні захисні прилади та алгоритми, які забезпечують додатковий захисний бар'єр на шляху несанкціонованого отримання інформації із захищеної системи. Приклади: сучасні смарт-карти зі спеціальними пристроями забезпечення безпеки; спеціалізовані інтегральні мікросхеми; захищені FPGA (field-programmable gate array) і CPLD (complex programmable logic device) мікросхеми.
6) *Високий рівень.* Система захищена від відомих типів атак, в тому числі деструктивних. Для знаходження уразливого місця системи потрібне масштабне дослідження з використанням спеціалізованого устаткування, при цьому невідомо чи можливо взагалі провести таке дослідження, і чи буде воно успішним. Прикладом є захищені апаратні криптографічні модулі.

У межах FIPS 140-2 (Federal Information Processing Standard) - американського та канадського стандарту обробки інформації [9], є чотири можливих рівні безпеки, одному із яких може відповідати система, пристрій або програма:

Рівень Безпеки 1 є найнижчим рівнем безпеки. Цей рівень визначає основні вимоги безпеки для криптографічних модулів.

Рівень Безпеки 2 вимагає покращення фізичної безпеки криптографічних модулів першого рівня, додаючи вимоги щодо опломбування, або стійких до зламу замків.

Рівень Безпеки 3 вимагає наявність додаткових засобів фізичної безпеки, які б перешкоджали зловмисникові одержати доступ до внутрішніх даних або параметрів системи або криптографічного модуля.

Рівень Безпеки 4 є найвищим рівнем безпеки. Вимогами даного рівня є забезпечення надійної захисної оболонки навколо криптографічного модуля, а

також наявність системи стеження для виявлення проникнення в пристрій ззовні.

Слід зазначити, що рівень безпеки певного пристрою не є величиною постійною. З часом може бути знайдений спосіб швидкого зламування системи, за допомогою більш нових та досконалих інструментів, або просто за допомогою нових методів атак, які будуть винайдені в майбутньому.

Основними шляхами можливого отримання інформації про внутрішній стан мікроконтроллера є: деструктивні методи, частково деструктивні та не деструктивні методи атак.

1.2. Деструктивні методи атак

Деструктивні методи атак вимагають наявності прямого доступу до внутрішніх компонентів пристрою [6]. Наприклад, якщо мова йде про апаратний USB-ключ, в цьому випадку потрібно відкрити корпус ключа для того, щоб отримати доступ до мікросхем пам’яті та мікропроцесорів, які знаходяться всередині корпуса такого ключа. У випадку старт-карти або мікроконтролера, потрібне руйнування упаковки (корпуса) мікросхеми, наприклад, за допомогою кислоти, та зняття верхнього слою кремнієвої пасивації для того, щоб отримати доступ до внутрішніх електричних з’єднань. На рис.1.1 показаний інструмент для попередньої підготовки мікросхеми до відкриття корпуса, а на рис.1.2. показаний корпус мікросхеми, попередньо підготовлений до відкриття за допомогою такого інструменту.

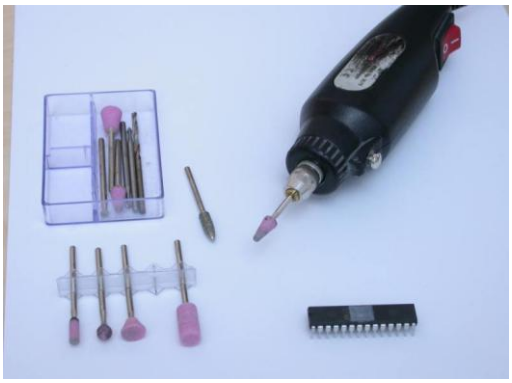


Рис. 1.1. інструменти для попередньої підготовки мікросхеми до відкриття корпуса

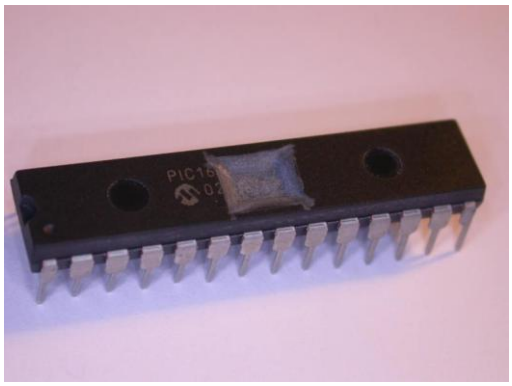


Рис. 1.2. підготовлений корпус

Відкриття корпуса мікросхеми – непросте завдання, яке потребує обережності, для того щоб не пошкодити сам кремнієвий кристал та тонкі провідники, за допомогою яких кристал під'єднується безпосередньо до виводів мікросхеми. Спершу треба вручну зробити невелике заглиблення в пластмасі якраз над тим місцем, де розташовано кристал напівпровідника. Робиться це для того, щоб азотна кислота HNO_3, яка буде використовуватися для розчинення пластмаси, з якої складається упаковка чіпа, розчинювала пластмасу в потрібному місці в районі кристала, і при цьому не розчинювала пластмасу в районі виводів або внутрішніх провідників мікросхеми. У якості розчинника для пластмаси в даному випадку використовується концентрована азотна кислота (концентрація більше 95%). Азотна кислота реагує з пластмасою, і внаслідок окислення та нітрації, пластмаса перетворюється на карбон. Крім того, азотна кислота впливає на мідь та срібло – матеріали з яких складається виводи мікросхеми та внутрішні електричні з'єднання між кристалом та зовнішніми виводами. Іноді, для прискорення реакції використовують суміш азотної кислоти та соляної кислоти (H_2SO_4). Це прискорює реакцію а також запобігає розчиненню виводів мікросхеми з срібла та міді (рис.1.3).

Реакція з використанням кислоти має проводитись при температурі 50-70 градусів, при цьому через кожні 10-30 секунд продукти реакції повинні видалятися за допомогою ацетону. Після того, як в отворі з’являєтья кремнієвий чіп, переходять до очищення кристала від продуктів реакції азотної кислоти з пластиком. Для цього мікросхему, змочену в ацетоні, поміщують у ультразвукову ванну. Після цього за допомогою повітродувки або фена здувають залишки хімічних реагентів. На цьому етап відкриття упаковки чіпа завершено (рис.1.4).

Наступним етапом дослідження є отримання знімків поверхні кристала за допомогою спеціального мікроскопа, та внесення змін в в кремнієву структуру або в металізацію кристала.

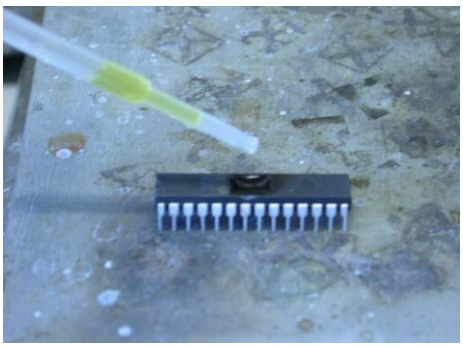


Рис.1.3. Розчинення пластмасового корпуса мікрочіпа за допомогою азотної кислоти.

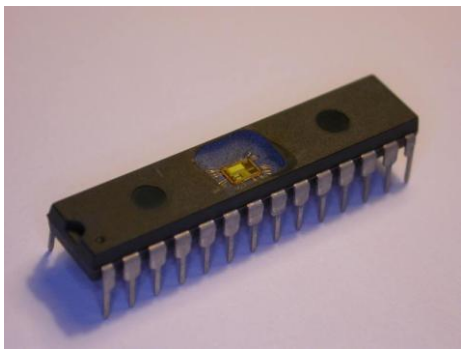


Рис.1.4. Мікрочіп з відкритим кристалом.

Необхідним обладнанням для даного типу атак є наявність станції для мікрозондування (рис.1.5, 1.6).

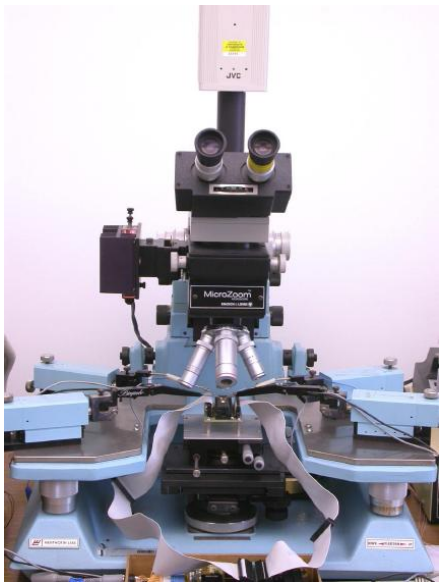


Рис.1.5. Мікроскоп для проведення ручного мікрозондування Wentworth Labs MP-901.

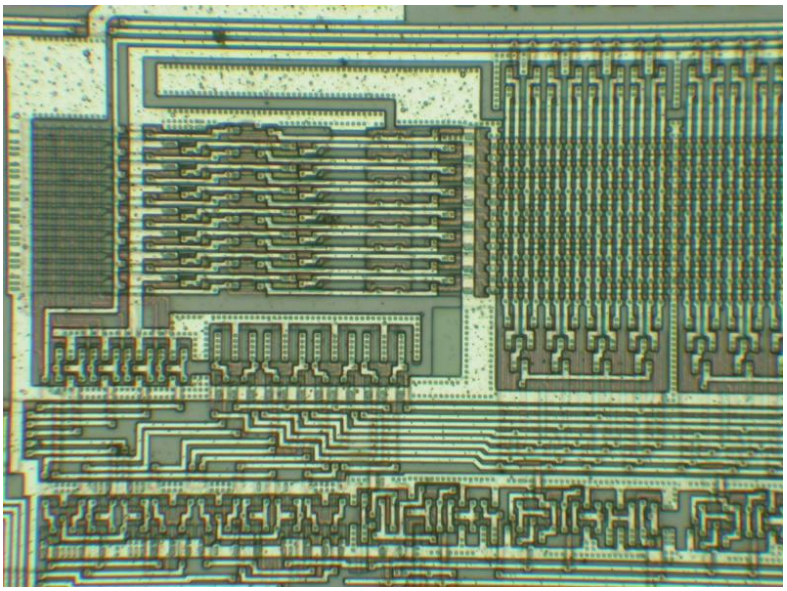


Рис.1.6. Знімок поверхні кристала, отриманий за допомогою мікроскопа.

Основним компонентом такої станції є спеціальний оптичний або електронний мікроскоп з великою між фокусною відстанню. Ще одним спеціальним компонентом є прецизійний зонд (щуп), який дозволяє приєднувати зондуючий контакт до електричних провідників досліджуваного кристала, точність встановлення зонда повинна бути менше мікрона.

Достоїнством деструктивних методів є їх необмежена можливість зчитування будь-якої інформації. Недоліком деструктивних методів є потреба у дорогому обладнанні, такому як електронний мікроскоп з лазером та прецизійною системою наведення.

Також, деструктивний метод передбачає вивчення структури кристалу, що є досить важкою процедурою, оскільки кількість транзисторів на кристалі у сучасних мікросхемах перевищує мільйони. Розташування транзисторів часто буває багатошаровим, що затрудняє вивчення структури кристалу та пошук необхідних для модифікації точок.

1.3. Частково деструктивні методи атак

Частково деструктивні методи є модифікацією недеструктивних методів з тією різницею, що немає потреби знімати верхній шар пасивації мікросхеми Теоретично частково деструктивні атаки можуть бути проведені з використанням таких засобів, як ультрафіолетове і рентгенівське випромінювання, лазери, електромагнітні поля і локальний нагрів. Вони можуть бути використані як окремо, так і в поєднанні один з одним.

За допомогою освітлення ультрафіолетом домагаються стирання окремих ділянок внутрішньої пам’яті або бітів захисту та відключають захист пам‘яті.

За допомого лазерне сканування поверхні кристалу мікросхеми отриманого знімка можна визначити стан кожного транзистора (включений чи виключений) та отримати відомості про стан мікроконтролера у даний момент.

Аналіз електромагнітного випромінювання краще проводиться на розпакованому чіпі. Стирання біта захисту в мікроконтролері шляхом опромінювання його ультрафіолетовим світлом теж вимагає розпаковування чіпа [10].

Лазерне випромінювання здатне іонізувати напівпровідникові області інтегральної схеми, якщо енергія його фотонів перевищує ширину забороненої зони напівпровідника. Наприклад випромінювання з довжиною хвилі 1,06 мкм (енергія фотонів 1.17еВ) дозволяє отримати глибину проникнення близько 700 мкм і дає змогу провести рівномірну просторову іонізацію кремнієвих пристроїв. Проте дисперсія обмежує його фокусування до декількох мікрометрів, що недостатньо для сучасних напівпровідникових пристроїв. Хоча при переході від інфрачервоного до видимого світла поглинання фотонів стрімко зростає, використання червоного і зеленого лазерів стало можливим оскільки транзистори в сучасних чіпах стали тоншими. Менший розмір пристроїв також означає, що для досягнення того ж рівня іонізації необхідно менше енергії.

У випадку з пристроями, виконаними за КМДН технологією (Комплементарна Метал Діелектрик Напівпроводник) існує небезпека ефекту насичення транзисторів, що викликає коротке замикання і що приводить до виходу пристрою з ладу. Тому використання радіації на КМДН структурах повинно проводитися з відповідною обережністю.

Розглянемо структуру елементу SRAM пам'яті, що містить дві пари p- і n-канальних транзисторів, що утворюють тригер, тоді як два інших n-канальних транзистора використовуються для читання його стану і запису в нього нових значень. Структурна схема елементу зображена на рис.1.7, а топологія - на рис.1.8. Транзистори Т1 і Т3 утворюють КМДН інвертор; разом з іншою подібною парою вони утворюють тригер, який контролюється транзисторами Т5 і Т6.

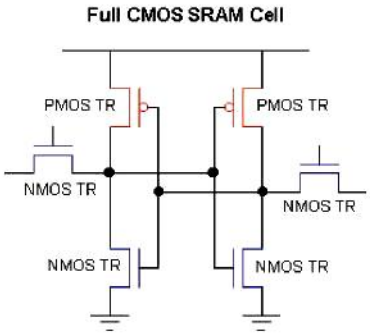


Рис.1.7. Структурна схема елементу статичного ОЗП

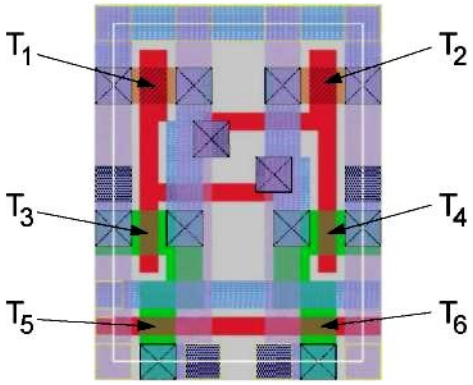


Рис.1.8. Топологія елементу статичного ОЗП

Якщо транзистор Т3 вдасться відкрити на короткий час зовнішнім імпульсом, це може привести до зміни стану тригера. Впливаючи на транзистор Т4, стан тригера може бути змінений на протилежний. Основна складність полягає у фокусуванні іонізуючого випромінювання в області декількох квадратних мікрометрів і вибору необхідної інтенсивності випромінювання.

Завдяки опроміненню вдається встановлювати в необхідний стан будь-який окремий біт в статичному оперативно-запам‘ятовувальному пристрою (ОЗП). Область ОЗП при максимальному збільшенні показана на рис.1.9. Фокусування світла від фотоспалаху на область обмежену білим колом, викликає перемикання комірки із стану лог.1 в стан лог. 0 або ніяких змін, якщо комірка була у стані 0. Фокусування світла на область, показану чорним колом, викликає перехід комірки із 0 в 1 або ніяких змін, якщо вона вже була в 1.

На рис.1.10 виразно видно, що матриця ОЗП розділена на дев’ять блоків, вісім з яких виглядають однаково. Опромінюючи комірки із різних областей, можна зробити висновок, що кожна область відповідає окремому біту даних записаної інформації. Експериментально було визначено, що лівий блок на рис 1.10 містить усі сьомі біти, другий зліва блок містить усі шості біти, і т.д., про що на рис 1.10 було нанесено відповідні позначки. Правий дев’ятий містить логіку адресації, і не відповідає безпосередньо за вміст пам’яті.

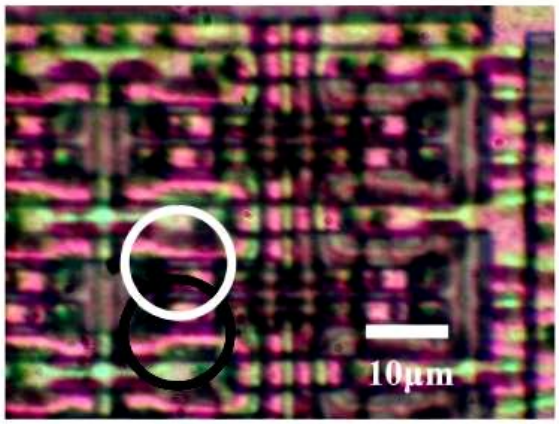


Рис.1.9. Матриця статичного ОЗП при максимальному збільшенні

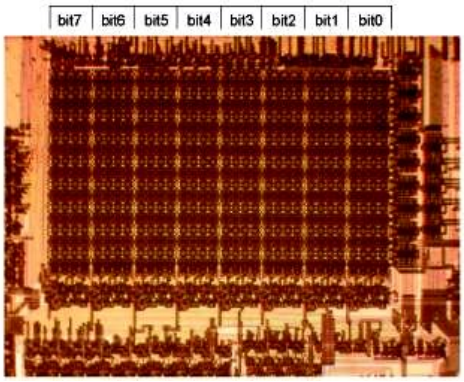


Рис.1.10. Розташування бітів даних у матриці ОЗП PIC16F84

Шляхом послідовного опромінювання кожного елементу в блоці побудована карта адрес бітів в адресному просторі пам'яті мікросхеми, що відповідають фізичному розташуванню кожної комірки в бітовому блоці на чіпі. Результат показаний на рис.1.11. Лівий край таблиці відповідає нижньому

краю кожного бітового блоку на рис 1.10. Можна також відмітити, що адреси розташовуються не послідовно, а розділені на три групи.

30h	34h	38h	3Ch	40h	44h	48h	4Ch	10h	14h	18h	1Ch	20h	24h	28h	2Ch	0Ch
31h	35h	39h	3Dh	41h	45h	49h	4Dh	11h	15h	19h	1Dh	21h	25h	29h	2Dh	0Dh
32h	36h	3Ah	3Eh	42h	46h	4Ah	4Eh	12h	16h	1Ah	1Eh	22h	26h	2Ah	2Eh	0Eh
33h	37h	3Bh	3Fh	43h	47h	4Bh	4Fh	13h	17h	1Bh	1Fh	23h	27h	2Bh	2Fh	0Fh

Рис.1.11. Розташування адреси в кожному блоці ОЗП PIC16F84

Таким чином частково деструктивні атаки можуть бути використані для відновлення карти пам'яті. Єдине обмеження полягає в тому, що фотоспалах не дає рівномірного і монохроматичного світла, тобто треба приділити особливу увагу області дії випромінювання. Це вирішується при застосуванні відповідного лазера.

В умовах, коли конструкція та принципи функціонування чіпа вже відомі, існує дуже потужна технологія, розроблена в IBM для дослідження чіпа в роботі навіть без видалення захисного шару. Для вимірювання робочих характеристик пристрою над ним поміщають кристал ніобата літію. Показник заломлення цієї субстанції змінюється при зміні електричного поля, і потенціал, що знаходиться під кремнієм може зчитуватися за допомогою ультрафіолетового лазерного променя, що проходить через кристал під ковзаючим кутом. Можливості цієї технології такі, що можна зчитувати сигнал у 5 В і з частотою до 25 МГц. По суті, це стандартний спосіб для добре оснащених лабораторій при відновленні криптоключів в чіпах, конструкція яких відома.

Андерсон і Кун вказали на те, що втручання в інструкції переходу процесора є потужною атакою: зломщик, який може викликати помилку в умовному переході при виконання програмного коду в старт-карті та, наприклад, зменшити число циклів в шифруванні, зробить отримання

секретного ключа досить простим завданням, що і підтверджує приведений вище приклад [1].

Описані вище атаки з випромінюванням є потужною технологією для атак на старт-карти і інші захищені процесори. Передбачається, що подібно до аналізу споживаної потужності запропонованому Кочером, вони можуть мати значний комерційний ефект для індустрії в тому, що потребуватимуть ретельної переоцінки вимог безпеки і введення нових технологій захисту [8].

У червні 2002 р. був оприлюднений метод злому смарт-карт і захищених мікроконтролерів, що отримав назву "атака оптичним індукуванням збоїв" (optical fault induction attack). Цей клас атак був виявлений і досліджений у Кембриджському університеті Сергієм Скоробогатовим та його керівником Росом Андерсоном [10]. Суть методу в тому, що сфокусоване освітлення конкретного транзистора в електронній схемі стимулює в ньому провідність, чим спричиняється короткочасний збій. Такого роду атаки виявляються досить дешевими і практичними, для них не потрібно складного і дорогого лазерного обладнання.

Для ілюстрації потужності нової атаки була розроблена методика, що дозволяє за допомогою фотоспалаху і мікроскопа виставляти в потрібне значення (0 або 1) будь-який біт у SRAM-пам'яті мікроконтролера. Методом "оптичного зондування" (optical probing) можна індукувати збої в роботі криптографічних алгоритмів або протоколів, а також вносити спотворення в потік керуючих команд процесора. Зрозуміло, що перераховані можливості істотно розширюють вже відомі методи злому криптосхем за збоями та вилучення секретної інформації з смарт-карт.

Індустрія, як зазвичай, намагається всіляко зменшити значущість нового методу атак, оскільки він відноситься до класу деструктивних атак, що супроводжуються пошкодженням захисного шару в чіпі смарт-карти. Однак, за свідченням Андерсона, зловмисники можуть обійтися і мінімальним фізичним втручанням: кремній прозорий в інфрачервоному діапазоні, тому атаку можна

проводити прямо через кремнієву підкладку із заднього боку чіпа, знявши лише пластик. Використовуючи ж рентгенівське випромінювання, карту і зовсім можна залишити незайманою.

Цими ж фахівцями з Кембриджа спільно з вченими комп'ютерної лабораторії Лувенського університету (Бельгія) нещодавно розроблені ще кілька нових методів зчитування інформації із захищених чіпів. Наприклад, скануючи чіп сфокусованим лазером (рис.1.12), або наводячи в ньому вихрові струми за допомогою індуктивності на голці мікропробника, можна підвищити електромагнітний витік, що видає записане там значення біта, але при цьому саме це значення зберігається в комірці непорушеним.

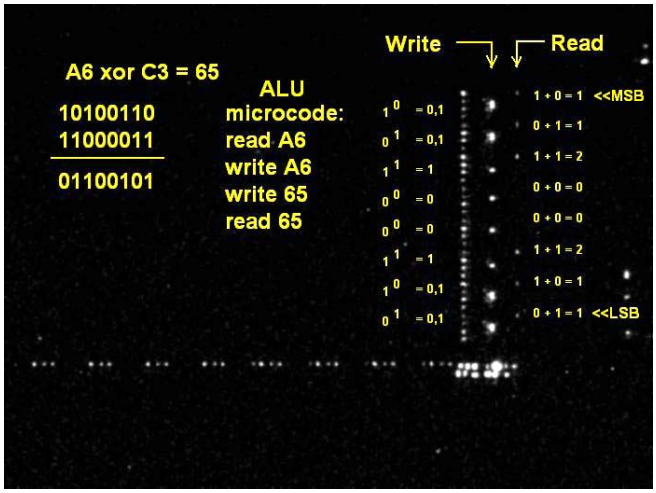


Рис.1.12. Результати сканування за допомогою інфрачервоного випромінення

Сильним охолодженням чіпа в потрібний момент часу можна "заморозити" вміст потрібного регістра і отримати з нього (ключову) інформацію, що зазвичай зберігається або передається в зашифрованому вигляді. Ця технологія застосовна до самих різних типів пам'яті від RAM до FLASH і реально продемонстрована зчитуванням ключів DES з пам’яті RAM без будь-якого фізичного контакту з чіпом.

Перевагою частково деструктивного типу атак є спрощений інструментарій (немає потреби в точному позиціонуванні та лазерному різаку). Недоліком є необхідність зняття корпусу мікросхеми, що може привести до порушення її структури, а також значна трудомісткість подібного дослідження у випадку неавтоматизованого вивчення стану кожного транзистора.

1.4. Недеструктивні методи атак на дані в мікропроцесорних системах

Недеструктивні методи дозволяють отримати інформацію про внутрішній стан мікроконтролера без зняття упаковки мікросхеми.

Недеструктивні методи атак включають у себе:

1) Дослідження часу виконання програми;
2) Атаки з повним перебором;
3) Виклик збоїв;
4) Аналіз побічних каналів витоку інформації;
5) Простий аналіз живлення;
6) Диференційний аналіз живлення;
7) Диференційний аналіз живлення високого порядку.

Дослідження часу виконання програми

Якщо програма у мікроконтролері обробляє великі масиви даних, існує можливість подаючи вхідні масиви даних різного розміру та наповнення, вимірювати час обробки цих даних, і тим самим знайти алгоритм кодування або перетворення інформації. Слід зазначити, що така можливість існує тільки для відомих алгоритмів. Якщо алгоритм перетворення інформації невідомий, вимірювання часу виконання програми не буде ефективним методом. Також, цей метод може бути неефективним при дослідженні мікроконтролерів з вбудованим RC-генератором [11]. У цьому випадку тактова частота задаючого генератора буде значно змінюватись зі зміною температури напівпровідникового кристала. Крім того, сучасні захищені мікроконтролери та смарт-карти можуть в процесі роботи самостійно змінювати тактову частоту,

що взагалі унеможливлює використання методу дослідження часу виконання програм.

Атаки з повним перебором. Якщо треба визначити пароль доступу до захищеної інформації, використовують повний перебір можливих комбінацій паролів та ключів [12]. Додатково, інформація про алгоритм перевірки пароля дозволяє значно скоротити час перебору. Наприклад, якщо відома певна частина паролю, або якісь обмеження при його перевірці, можливо певним чином змінити алгоритм перебору так, щоб скоротити число комбінацій та час підбору. Даний вид атаки має сенс лише при невеликій кількості комбінацій паролю. Окрім того, програма може блокувати прийняття пароля після третьої спроби або навмисно вводити часову затримку при перевірці пароля, тим самим значно ускладнюючи його підбір. Ще одним варіантом методу атак з повним перебором є подавання на всі контакти контролера різноманітних комбінацій сигналів з метою входження в режим покрокового виконання або відлагодження. Ці режими дозволяють зчитувати оперативну пам’ять контролера та регістри, що дозволяє повністю розкрити алгоритм кодування чи захисту.

Виклик збоїв. Відомо, що при виникненні збою в обчисленнях комп'ютерний пристрій може видати інформацію, корисну для відновлення секретних даних. До інженерно-захищених пристроїв типу смарт-карт, з метою виклику помилок при обчисленнях, застосовують певні рівні радіаційного опромінення або нагрівання, подачу неправильної напруги живлення або нестандартну тактову частоту [13] [10].

Наприкінці вересня 1996 колектив авторів з Bellcore, науково-дослідного центру американської компанії Bell, повідомив про те, що виявлена серйозна потенційна слабкість загального характеру в захищених криптографічних пристроях, зокрема, в смарт-картах для електронних платежів. Автори - Боне, Де-Мілль і Ліптон - назвали свій метод взлому "криптоаналіз при збоях обладнання" [12]. Суть методу в тому, що штучно викликаючи помилку в

роботі електронної схеми за допомогою іонізації або мікрохвильового опромінення, а потім порівнюючи значення при збої на виході пристрою із правильними значеннями, теоретично є вірогідність відновити криптографічну інформацію, що зберігається в смарт-картці.

Дослідження показали, що даній загрозі піддаються всі пристрої, що використовують криптоалгоритмами з відкритими ключами для шифрування інформації і аутентифікації користувача. Це можуть бути смарт-карти, що застосовуються для зберігання даних (наприклад, електронних грошей); SIM-картки для стільникової телефонії; картки електронного підпису та аутентифікації користувача при віддаленому доступі до корпоративних мереж. Розроблена в Bellcore атака була застосовна для отримання ключів виключно в криптосхемах з відкритим ключем – RSA (Rivest, Shamir, Adleman), алгоритму цифрового підпису Рабіна, схемі ідентифікації Фіата-Шаміра і тому подібних конструкції [5].

Головним результатом публікації Bellcore стало те, що до відомої проблеми було привернуто увагу набагато більшої кількості дослідників. Менше ніж через місяць після появи статті Боне та його колег, у жовтні 1996р., стало відомо про розробку аналогічної теоретичної атаки відносно симетричних шифрів, тобто криптоалгоритмів розкриття даних із загальним секретним ключем. Новий метод був розроблений знаменитим тандемом ізраїльських криптографів Елі Біхамом і Аді Шаміром, отримавши назву "Диференціальний аналіз спотворень" (ДАС) [2].

На прикладі найпоширенішого блокового шифру DES (Data Encryption Standard) ці автори продемонстрували, що в рамках тієї ж "беллкорівської" моделі, при виникненні збою в роботі апаратури є вірогідність визначити повний ключ DES з захищеної смарт-картки шляхом аналізу менше ніж 200 блоків шифротексту (один блок DES становить 8 байт). Більш того, згодом з'явився ще ряд робіт Біхама-Шаміра з описом методів вилучення ключа із

смарт-карти в умовах, коли про реалізацію криптосхеми не відомо практично нічого.

Навесні 1997 року з'явився опис не теоретичної, а досить практичної атаки, що отримала назву "удосконалений метод ДАС". В основу нового методу була покладена модель примусових спотворень або "глітч-атак" (від англійського glitch - сплеск, викид), що практикуються хакерами при зломі смарт-карт платного телебачення. Під глітч-атаками розуміються маніпуляції з тактовою частотою або напругою живлення смарт-карт, що дозволяє видавати данні з ключовим матеріалом на вихідний порт пристрою. Ефективність глітч-атак продемонстрована кембриджськими авторами як на симетричних криптосхемах, так і на алгоритмах з відкритим ключем [6].

Для виконання атак з викликом збоїв, під час роботи контролера змінюють певні параметри зовнішніх впливів, такі як: збільшення або зменшення робочої частоти контролера за межі штатного діапазону, збільшення або зменшення напруги живлення контролера, збільшення температури, введення тимчасових провалів або завад до напруги живлення або тактових сигналів контролера. Результатом виконання таких впливів може бути порушення виконання інструкцій контролера, що може призвести до збою в роботі внутрішньої програми та, як результат, до викриття певних захищених даних або ключів шифрування. Прикладом такої уразливої системи є мікроконтролер з пам’яттю EEPROM (Electrically Erasable Programmable Read-Only Memory). При зниженні напруги нижче певного рівня, не всі комірки такої пам’яті піддаються перепрограмуванню. Тому можливе спотворення записаної у пам’яті інформації. При наступному включенні контролера зчитана інформація буде невірною, що може призвести до перешкод у роботі мікропроцесорної системи. Виходом з даного становища є використання так званих Brown-out детекторів. Brown-out детектор – це пристрій, який містить порогові елементи, які відслідковують значення напруги живлення та зупиняють роботу контролера у випадку невідповідності поточного значення напруги, значенню, при якому

можлива стійка робота контролера та запис EEPROM пам’яті. Більшість сучасних мікроконтролерів містить вбудований Brown-out детектор, що дозволяє усунути проблеми з помилковими даними у пам’яті EEPROM.

Аналіз побічних каналів витоку інформації. В 1998 р. був запропонований новий метод розкриття конфіденційної інформації [5], успішно реалізований на практиці. Невелика консалтингова криптофірма Cryptography Research з Сан-Франциско розробила надзвичайно ефективний аналітичний інструментарій для видобування секретних ключів з криптографічних пристроїв. За словами глави фірми Пола Кочера, якому на той час було 25 років, дослідникам "не вдалося знайти жодної смарт-картки, яку не можна було б розкрити запропонованим методом".

У традиційному аналізі криптопристроїв і захищених протоколів прийнято припускати, що вхідне і вихідне повідомлення доступні зловмисникові, а невідома інформація знаходиться всередині (ключі шифрування). Однак, електронні пристрої складається з конкретних елементів, що видають у навколишнє середовище інформацію про свою роботу. А це, насправді, означає, що атакуючій стороні може бути доступна і інша побічна інформація, що видається криптопристроєм: електромагнітне випромінювання, сигнали про помилки або про інтервали часу між виконуваними інструкціями, коливання в споживанні електроживлення та інші данні.

Дані методи аналізу заслуговують проведення ґрунтовних досліджень, оскільки атаки такого типу проводяться швидко, крім того для виконання атак використовується вже готове обладнання вартістю від сотень доларів.

Більшість сучасних криптографічних пристроїв виконані на напівпровідникових логічних елементах, побудованих з транзисторів. Електрони, що протікають через кремнієву підкладку, коли до транзисторного затвору прикладений заряд, призводять до споживання струму і спричиняють електромагнітне випромінювання.

Криптографічні дослідження успішно використовують ці особливості для аналізу великого числа пристроїв, заснованих на смарт-картах. В той час, як деякі пристрої стійкі до простого аналізу живлення (Simple Power Analysis - SPA), дуже мало пристроїв, що серійно випускаються, протистоять диференціальному аналізу напруги (Differential Power Analysis - DPA). Кількість часу, необхідна для атаки залежить від типу атаки (DPA, SPA, і.т.д.) і від самого пристрою. SPA атаки зазвичай займають декілька секунд на картку, а DPA атаки можуть займати декілька годин. Базові концепції нової методики дослідження сформульовані у роботі [5], де показано, що існує вірогідність відкриття секретних даних у криптопристроях, просто точно заміряючи інтервали часу, які тим потрібні на обробку даних. Криптографічні пристрої використовують секретний ключ для обробки вхідної інформації і/або для надання вихідній інформації. Розробники протоколів припускають, що вхідна і вихідна інформація відомі, а інформація про ключ не відома.

Насправді можливість атак аналізом за струмом споживання а також споріднених атак, розроблені Полом Кочером і Cryptography Research, включаючи синхронні атаки і диференціальний аналіз живлення, що використовує електромагнітне випромінювання визначаються доступністю й іншої інформація (рис.1.13).

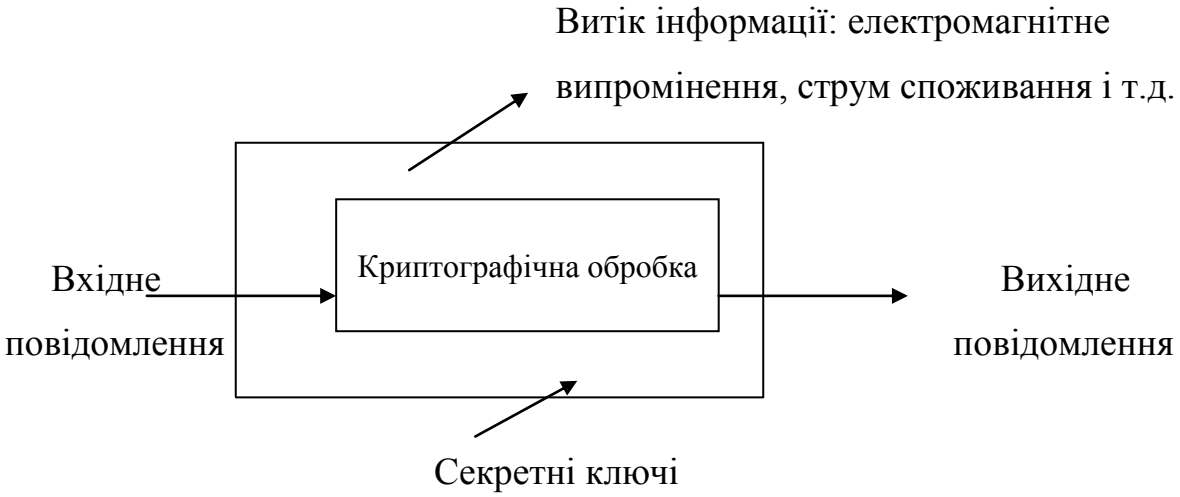


Рис.1.13. Ілюстрація процесу криптографічної обробки інформації

На рис 1.13 схематично відображений процес криптографічної оброки інформації. Вхідне повідомлення поступає на вхід захищеного пристрою, що містить блока криптографічної обробки та секретні ключі. Перетворення повідомлення полягає у його шифруванні або дешифруванні. У випадку шифрування повідомлення, на вхід блока подається відкрити текст, а на виході блока з'являється зашифрований текст. У випадку виконання алгоритму дешифрування, на вхід блока подається зашифрований текст, а на виході блока з'являться початковий відкритий текст. У реальний системах процес криптографічної обробки шифрування або дешифрування може супроводжуватися витоком інформації через електромагнітне випромінення, струм споживання та інші канали.

Вихідні дані, які необхідні для аналізу захищеності криптографічного пристрою, можуть бути досить точно виміряні за допомогою простих вимірювальних приладів. Зокрема, простій амперметр, сконструйований з активного навантаження та цифрового осцилографа може бути використаний для вимірювання струму споживання на виводах живлення.

Простий аналіз струму живлення . У атаках простим аналізом живлення, криптоаналітик безпосередньо досліджує струм споживання системи. Кількість споживаної потужності змінюється залежно від виконуваних мікропроцесором інструкцій. Великі обчислення, такі як DES раунди, RSA операції і т.д. можуть бути ідентифіковані, оскільки інструкції виконувані мікропроцесором значно змінюються протягом вказаних операцій.

Наведений рис.1.14 ілюструє SPA спостереження споживання струму i_{CC} при простій DES операції, виконаний типовою смарт-картою. Верхня крива показує струм споживання Icc при виконанні операції шифрування, включаючи початкове перемішування, 16 DES раундів, і кінцеве перемішування. Нижня крива більш детально відображає струм споживання при виконанні алгоритму шифрування у 2-му і 3-му раундах.

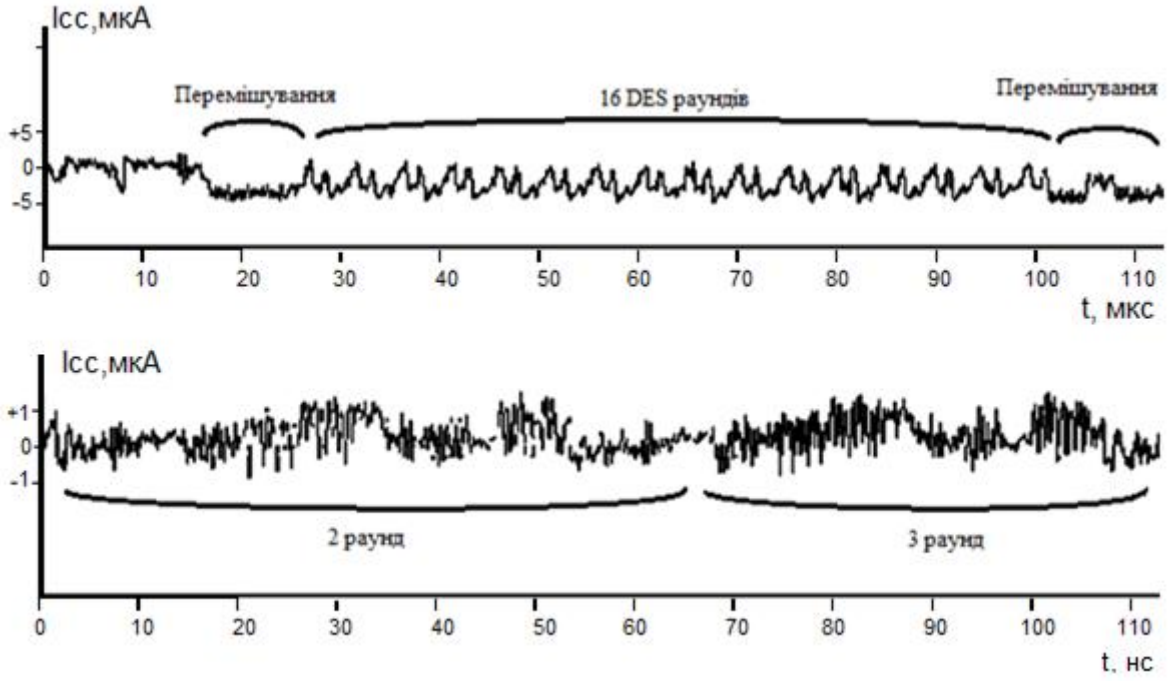


Рис.1.14. Діаграми струмів операцій шифрування, отримані при проведенні SPA-дослідження.

При великому коефіцієнті підсилення можуть бути помітні окремі інструкції. Простий аналіз струму живлення використовується, наприклад, для злому RSA реалізацій за допомогою виявлення відмінностей між операціями множення і обчислення квадратного кореня. Аналогічно, багато DES реалізацій мають видимі відмінності в перемішуваннях і зсувах, і можуть таким чином бути зламані, використовуючи SPA.

Диференціальний аналіз струму живлення. Диференціальний аналіз струму живлення є потужнішою атакою, ніж SPA, і складнішою для запобігання [6]. В той час, як SPA атаки головним чином будуються на візуальному аналізі з метою виділення значущих флуктуацій живлення, DPA атака використовує статистичний аналіз і техніку корекції помилок для виділення інформації, що має кореляції з секретними ключами. Для

використання статистичної техніки аналізуються результати декількох тисяч транзакцій.

На рис.1.15 показані чотири криві струму, отримані при використанні введення відкритого тексту в DES функцію шифрування в смарт-карті.

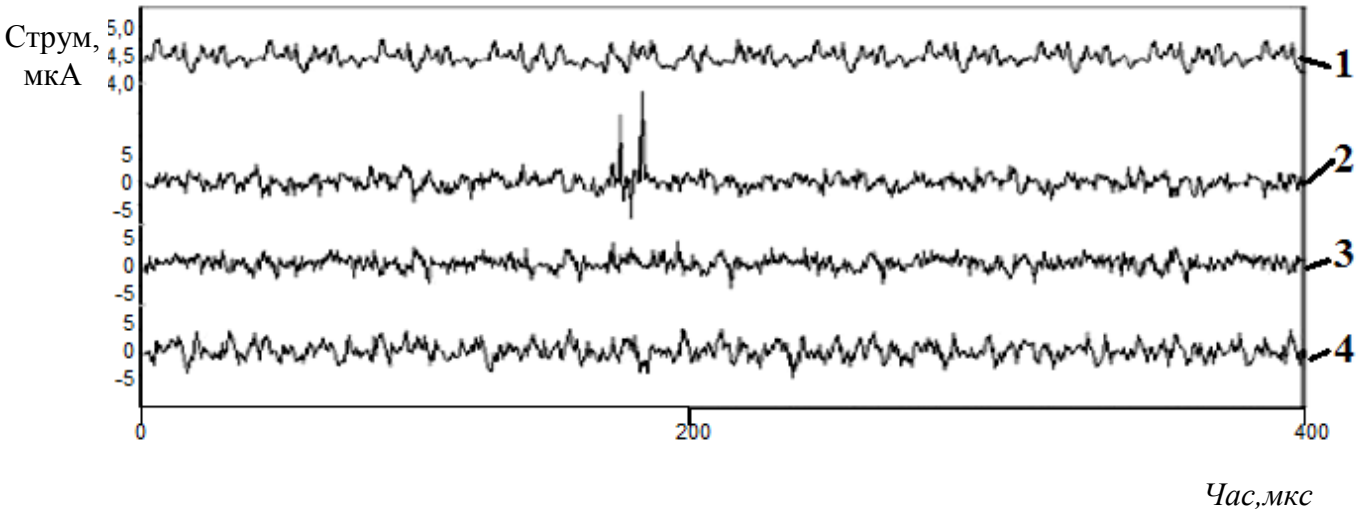


Рис.1.15. Струми споживання смарт-карти під час DES операцій для правильного і двох неправильних ключів.

Крива 1 (рис.1.15) зображає середнє споживання струму під час DES операцій. Нижче зображені три диференціальних кривих, де крива 2 представляє використання правильного варіанту значення ключа шифрування. Криві 3 та 4 відображають використання некоректного значення для ключа шифрування. Ці залежності усереднені з використанням 1000 вимірів. В сигналах присутній значний рівень шуму, однак самі сигнали виразно видно.

Диференціальний аналіз струму живлення високого порядку. Диференціальний аналіз струму живлення високого порядку являє собою ще складніший метод аналізу - HO-DPA (High Order Differential Power Analysis). Тоді як DPA-техніка аналізує інформацію між зразками даних впродовж окремої події, метод аналізу диференціалів високого порядку

використовується для кореляції інформації між багатьма криптографічними субопераціями.

У DPA атаці високого поряду сигнали збираються з багатьох джерел, використовуючи техніку різних вимірювань. Збираються сигнали з різними часовими затримками під час застосування техніки DPA, застосовуються велика кількість диференціальних функцій, а також спеціалізовані сигналообробниі функції. Основні HO-DPA обчислювальні функції є більш загальною формою стандартних функцій DPA.

На сьогоднішній день HO-DPA представляє великий інтерес для розробників систем і дослідників, оскільки не відомо реально існуючих систем, уразливих для HO-DPA, але не уразливих для DPA. В загальному випадку, HO-DPA вважається ефективнішим методом атаки, ніж DPA.

Техніка аналізу струму живлення є досить потужним видом атак, оскільки дуже велика кількість пристроїв, що використовується вразливі до цієї техніки. Атаки легкі для виконання, мають низьку вартість, і проводяться без руйнування мікросхем, що робить їх виявлення та протидію складним завданням. Найбільшу ефективність мають атаки DPA та HO-DPA. Крім того, виконання атаки за струмом споживання може бути автоматизоване.

1.5. Методи захисту інформації від зчитування за струмом споживання

На основі огляду наведених літературних джерел по методам захисту інформації від аналізу струму споживання пропонується класифікація, наведена на рис.1.16. Системи захисту від атак за струмом споживання поділяються на три основні типи:

1) системи, що зменшують корисний сигнал, який може бути доступним через виводи живлення;

2) системи, що вносять додатковий струм споживання, маскуючи при цьому корисний сигнал;

3) систем, що забезпечують непряме живлення мікроконтролерів, які у свою чергу поділяються на:

а) з тимчасовим підключенням (живлення подається на певному етапі роботи мікроконтролера або відповідно до потреб програмного забезпечення);

б) з неперервним підключенням (працюють протягом всього часу роботи мікроконтролера).

Наведені системи захисту за реалізацією поділяються на 1) програмні, 2) апаратні та 3)програмно-апаратні. При програмній реалізації, обробка захищених даних виконується з розподілом в часі або супроводжується хибними розрахунками. Апаратні системи базуються на згладжуванні струму споживання та внесенні додаткового струму споживання за рахунок використання нелінійних елементів та фільтрів. Програмно-апаратні системи захисту комбінують переваги як фільтрів, так і програмних алгоритмів.

Системи захисту першого типу, які зменшують корисний сигнал, можуть бути реалізовані на основі фільтрів або стабілізаторів.

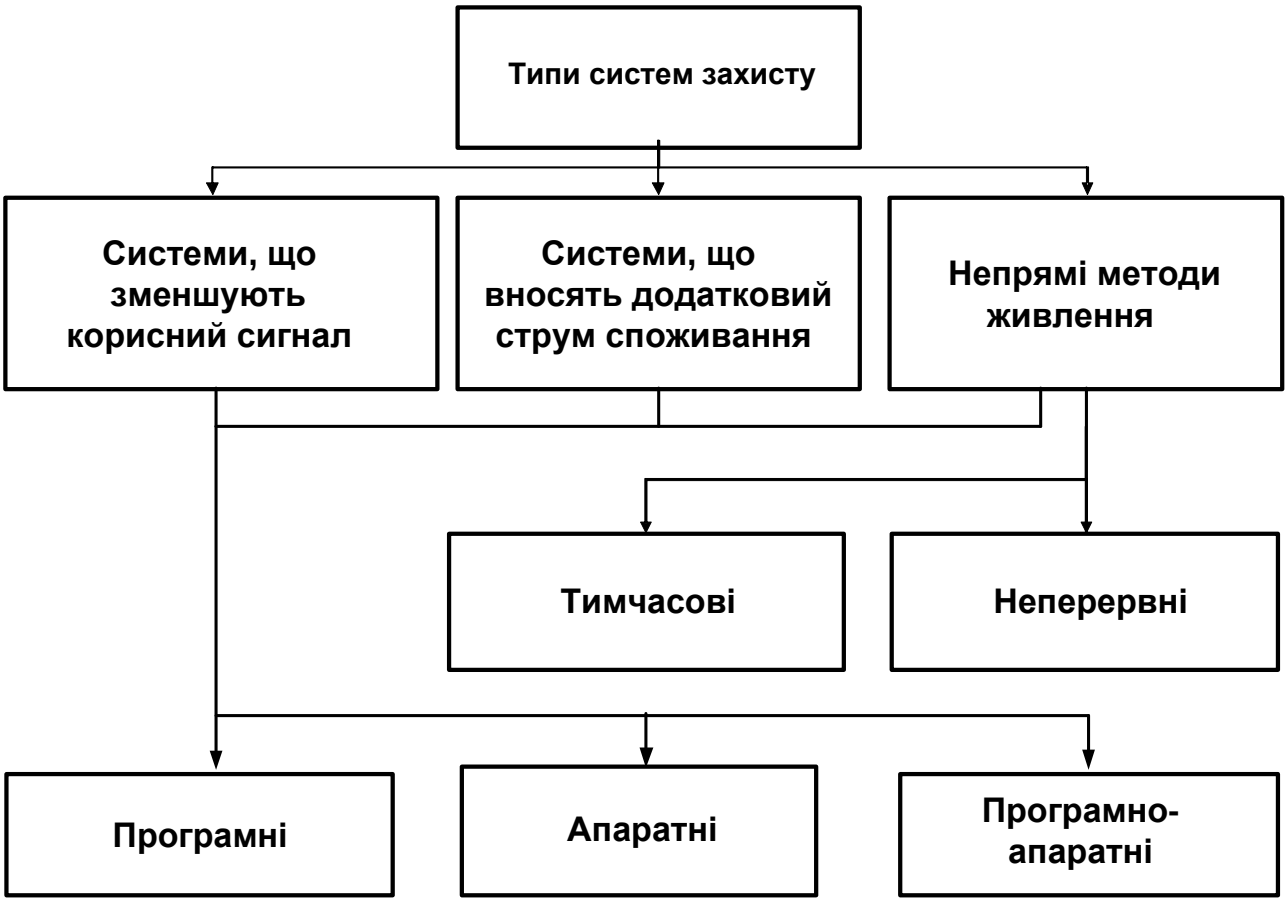


Рис.1.16. Основні типи систем захисту від атак за струмом споживання

Найпростіша система захисту першого типу [14] являє собою згладжуючий RLC фільтр (рис.1.17). За допомогою ключа S відбувається підключення конденсатора C, що змінює параметри фільтра.

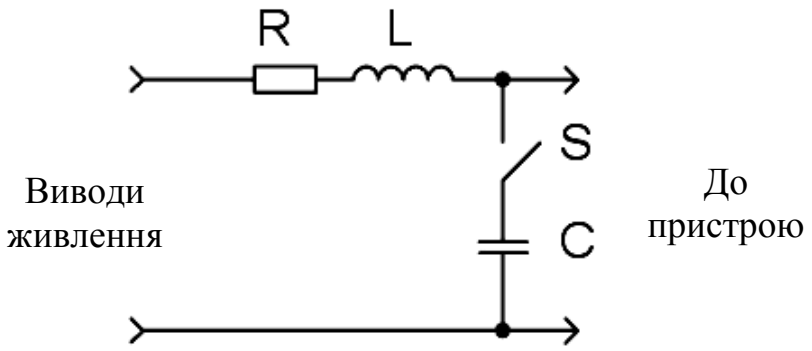


Рис.1.17. Система захисту, що зменшує корисний сигнал, на основі фільтру зі змінними параметрами.

Проходження струму поживання через реактивні компоненти фільтра дозволяє зменшити сплески струму та напруги та досягти певного зсуву фаз між ними.

Модифікаціями такої системи є комбіновані системи з використанням послідовного та паралельного з'єднання фільтрів, а також введення зворотних зв'язків (рис.1.18 а) або матричного включення [15] (рис.1.18 б).

При цьому керування такою структурою здійснюється від таймера або генератора випадкових чисел. Подібна система захисту забезпечує не тільки згладжування струму споживання, а й внесення в нього додаткових флуктуацій за рахунок перехідних процесів підключення та відключення ємності, співвідношення між параметрами ємності та індуктивності визначає тип та тривалість перехідного процесу. Складність реалізації цієї системи полягає у проблемах інтегрального виконання індуктивностей та ємностей для забезпечення суттєвого впливу на імпульсну складову струм споживання мікроконтролерів.

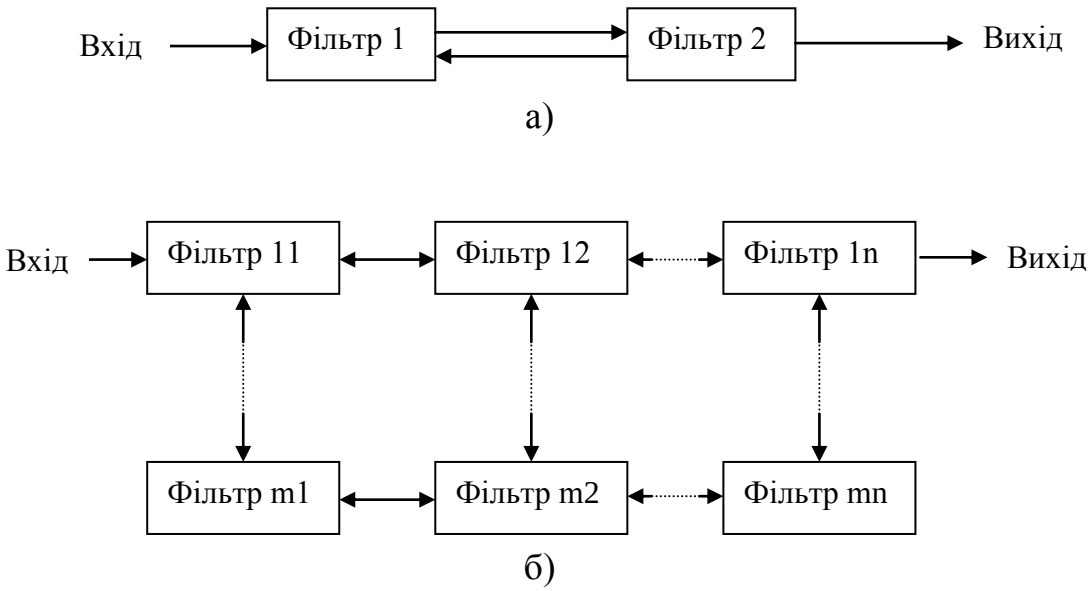


Рис.1.18 Структурна схема включення: а) декількох фільтрів зі зворотним зв'язком; б) матричне включення фільтрів.

Досить проста система захисту [16] [17] першого типу може бути виконана на базі фільтруючого накопичувального конденсатора С великої ємності (рис.1.19) та поєднувати в собі програмний та апаратний захист. Вхідний ємнісний фільтр, під’єднаний між виводами живлення VCC та GND забезпечує фільтрацію струму споживання та дозволяє підтримувати роботу вузлів мікроконтролера, таких як ЦП (центральний процесор), пам'ять, таймер та пристрої процесора деякий час після відключення живлення, щоб виконати операцію стирання пам’яті у випадку виникнення збою при виконанні обчислень над захищеними даними.

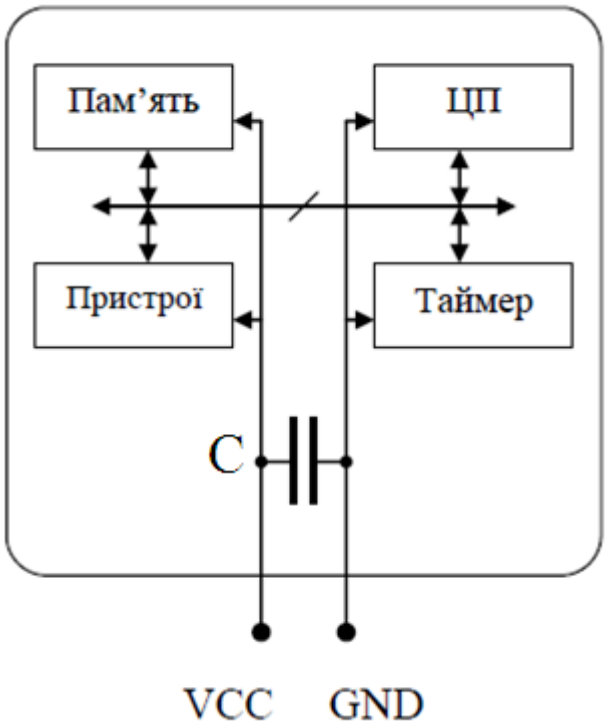


Рис.1.19. Структурна схема включення системи захисту на основі вхідного конденсатора.

Значно зменшити «корисний сигнал» струму споживання дозволяє система захисту [18] на основі нелінійних елементів (рис.1.20). Використання стабілітрона та транзистора, в активному режимі, підключених до джерела живлення послідовно з процесором, суттєво зменшують пульсації напруги. Це

потребує збільшення чутливості обладнання для проведення атаки за струмом споживання, а отже і його вартості.

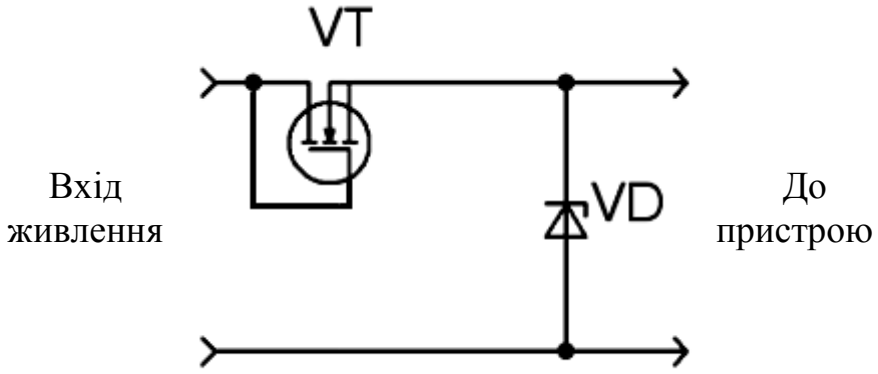


Рис.1.20. Система захисту на основі стабілізатора напруги.

Ця система розрахована на досягнення такого рівня складності детектування стандартних підпрограм, при якому проведення атаки було б економічно недоцільним або/і технічно складним.

Системи захисту другого типу, що вносять додатковий струм споживання, маскуючи при цьому «корисний сигнал», характеризується наявністю інтегрованого блоку, що дає можливість накладання на реальний струм споживання додаткового струму. І хоча сумарний струм споживання мікроконтролера дещо збільшується, вдається приховати виконання мікрокоманд, чим і забезпечується захист від атаки за струмом споживання. Основою таких систем є генератор випадкових чисел (ГВЧ), тип і конструкція якого в цілому визначає як величину, так і рівень хаотичності генерованого шуму.

Система захисту [19], структурна схема якої наведена на рис.1.21, базується на основі параметричного джерела струму. Живлення мікроконтролера здійснюється частково від джерела струму I_{CC}, I_N, а частково від накопичуючого конденсатора С, що заряджається при струмі споживання мікроконтролера, нижчому за вихідний струм джерела живлення. Такий метод дозволяє відбирати від джерела живлення, підключеного до виводів VCC, GND

сумарний струм споживання, обумовлений зарядом конденсатора С. Недоліком даної системи є її інерційність, і тому високочастотні сплески струму живлення ЦП все одно будуть проходити до джерела живлення всієї системи.

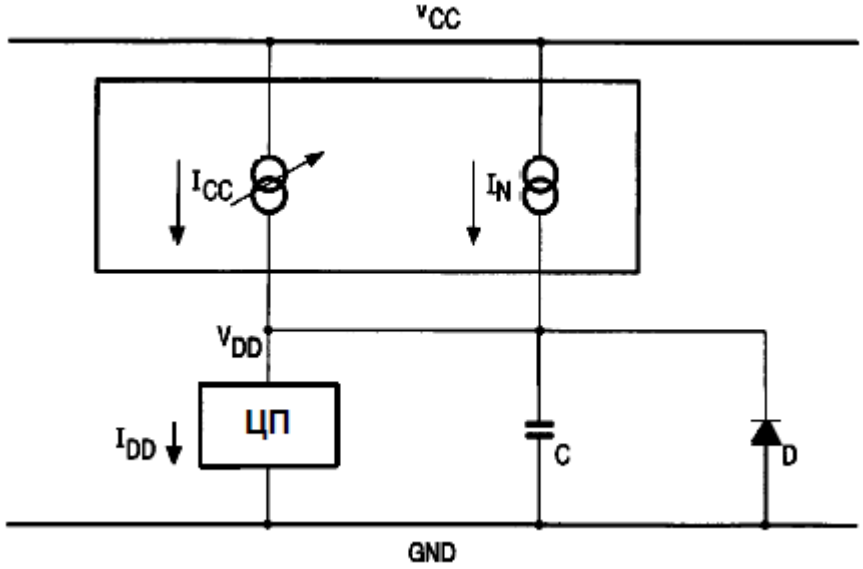


Рис.1.21. Система захисту на основі параметричного джерела струму.

Відомою є система захисту на основі блоку ключів [20] [21], структурна схема якої приведена на рис.1.22.

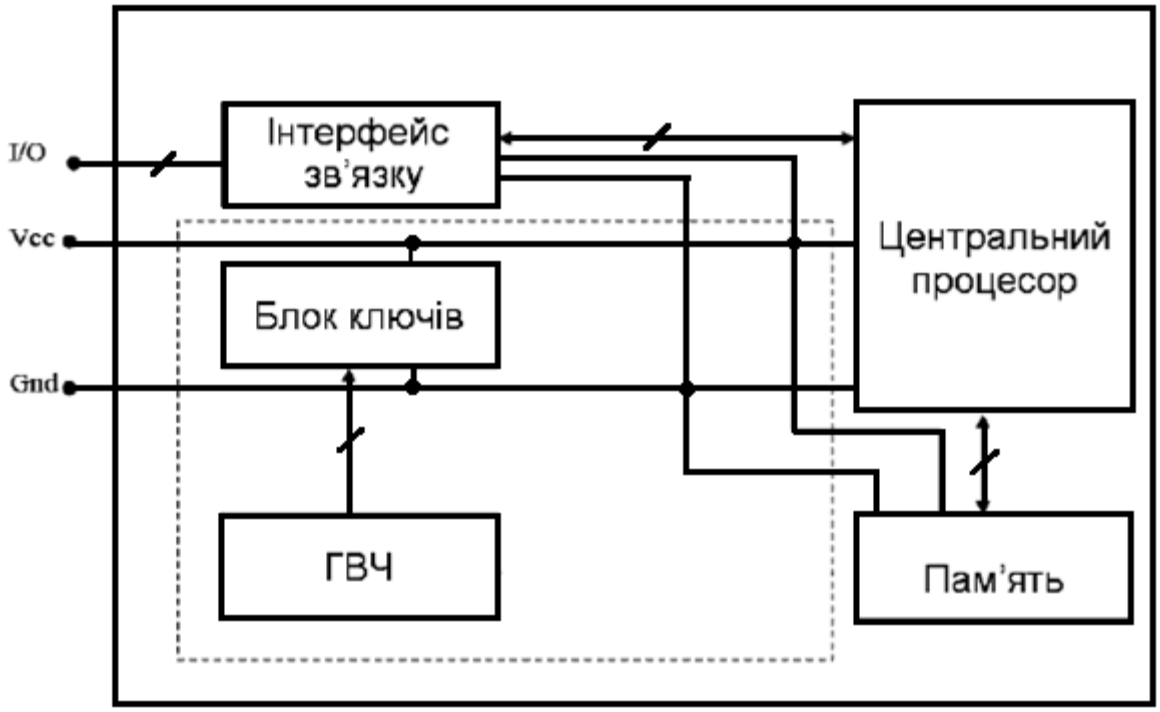


Рис.1.22. Структурна схема системи захисту на основі блоку ключів.

Система керування на основі ГВЧ, що виконується на основі шумлячого діода представляє вихідний шум як бінарну послідовність, що надходить на виводи керування ключами та забезпечує їхнє ввімкнення-вимкнення. Паралельно до виводів живлення мікроконтролера VCC та GND під'єднується блок ключів, комутації у якому супроводжуються споживанням струму при перехідних процесах, та додають певний рівень постійної складової струму. Зв'язок мікроконтролера із зовнішніми пристроями проводиться по виводам введення-виведення I/O за допомогою інтерфейсу введення-виведення. В мікроконтролері також міститься центральний процесор, який виконує криптографічну обробку інформації та пам'ять, яка містить секретний ключ та оброблювані дані. Виводи живлення центрального процесора та пам'яті мають обов'язково підключатися паралельно до блоку ключів. Залежно від системи керування блок ключів може виконуватися з однаковими опорами, під'єднаними до кожного ключа, або із неоднаковими опорами, що дає можливість змінювати додатковий струм споживання в широкому діапазоні. Реалізація такої системи захисту спричиняє нераціональне використання площі кристалу та необхідності розробки додаткових аналогових і цифрових інтегральних пристроїв для здійснення керування.

Системи захисту третього типу являють собою непрямі системи живлення мікроконтролерів, що додатково можуть містити попередньо розглянуті структури для збільшення ступеня захищеності. Найпростіші системи такого типу (рис.1.23 а-в) призначені для тимчасового від'єднання внутрішніх елементів мікроконтролера від зовнішнього джерела живлення, функціонування яких підтримується за рахунок накопиченої енергії. Система захисту [22], що зображена на рис.1.23 а., має вхідний ключ на транзисторі, що перемикає живлення мікроконтролера CPU між накопичуючим конденсатором та зовнішнім джерелом VCC, GND. Перемикання ключа керуючим сигналом по виводу CTL здійснюється з певним періодом або згідно підпрограми, що виконується.

Пристрій захисту [23] на рис.1.23 б є вдосконаленням попередньої системи і характеризуються ускладненням структури комутованих конденсаторів та алгоритму їх роботи. Живлення центрального процесора ЦП подається з виводів VCC та GND за допомогою ключа, по заданій програмі. Одним із варіантів побудови алгоритму перемикання ключа є відключення живлення від центрального процесора лише під час виконання важливих операцій із секретними даними, що дозволяє суттєво зменшити час, протягом якого можливе проведення атаки. Інтегровані пристрої в даній схемі – це пристрої, зчитування струму споживання яких не робить критичною захищеність мікропроцесорної системи в цілому.

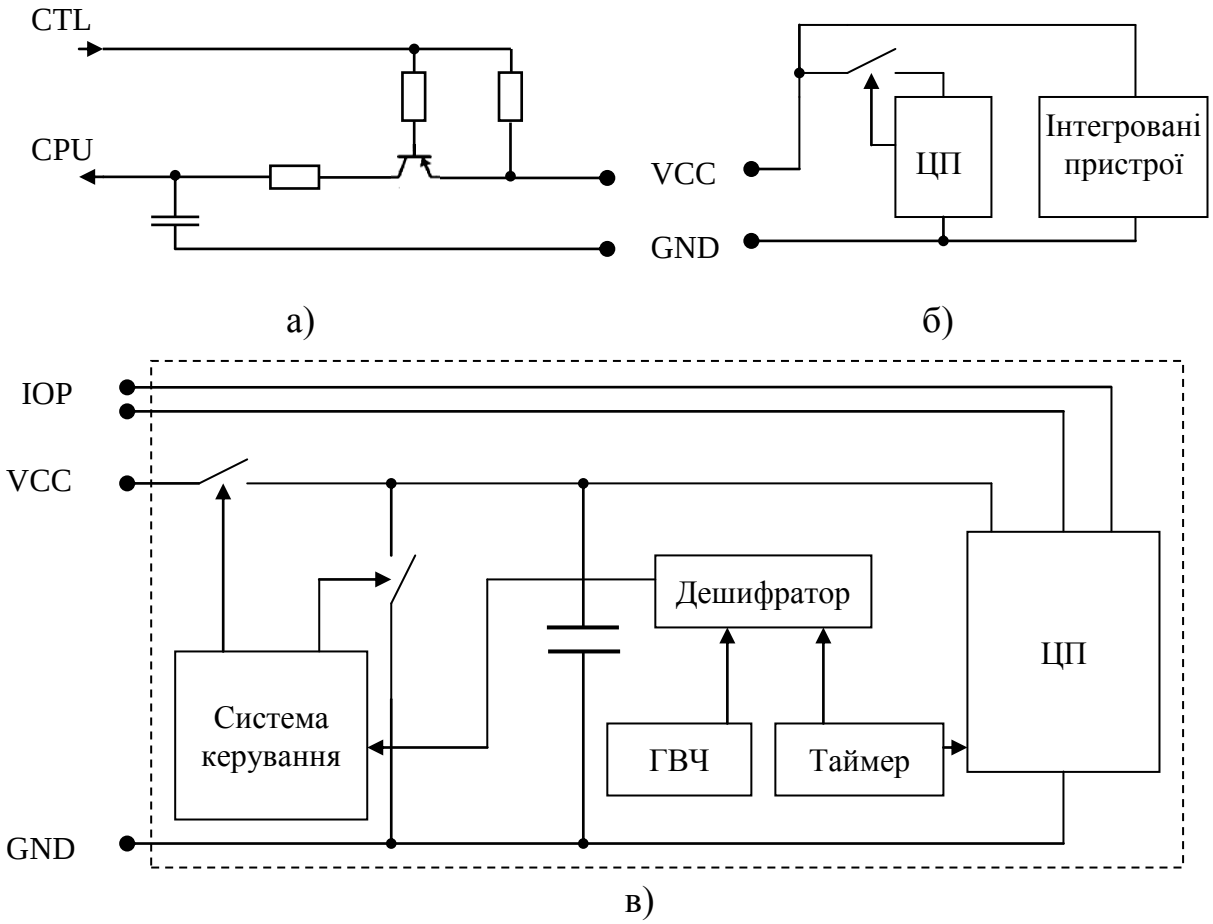


Рис.1.23. Типові непрямі системи живлення мікроконтролерів.

Більш гнучка система захисту [24] зображена на рис.1.23 в. У цій системі центральний процесор (ЦП) виконує криптографічні операції, і зчитування його

струму споживання небажане. Обмін інформацією між центральним процесором та зовнішніми пристроями здійснюється за допомогою портів введення-виведення (IOP). Система керування ключами здійснює комутацію напруги живлення з виводів VCC та GND, і виконує заряд або розряд конденсатора. Вказаний конденсатор також використовується для тимчасового живлення ЦП у той час, коли зовнішнє джерело живлення відключене. Таймер задає максимальне значення часу живлення мікроконтролера від конденсатора, ГВЧ робить час живлення неоднаковим у кожному наступному випадку, а дешифратор узгоджує сигнали таймера та ГВЧ для створення необхідних імпульсів керування системою захисту.

Існуюча система захисту із непрямим методом живлення центрального процесора (рис.1.24) побудована на основі незалежного джерела живлення на двох конденсаторах [25]. Виводи живлення ЦП через ключі S1-S4 під'єднано до конденсаторів C1, C2 з ємністю, достатньою для безперебійної тимчасової роботи ЦП впродовж декількох тактів. За рахунок переключення ключів, живлення центрального процесора та периферійних пристроїв постійно відбувається від одного з конденсаторів в той час як інший заряджається від зовнішнього джерела.

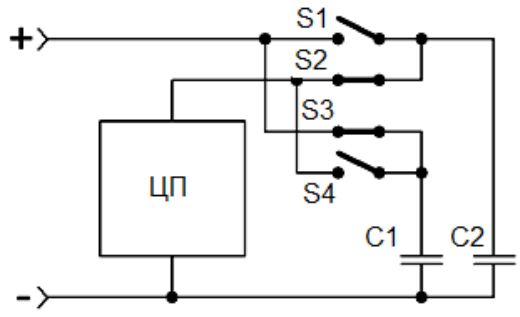


Рис.1.24. Джерело живлення мікроконтролера
на основі двох конденсаторів.

Наведені системи захисту можуть бути виконані в розподіленому вигляді (рис.1.25) з однотипних ланок, що змінюють струм споживання та включаються окремо для кожної складової мікроконтролера, що потребують захисту.

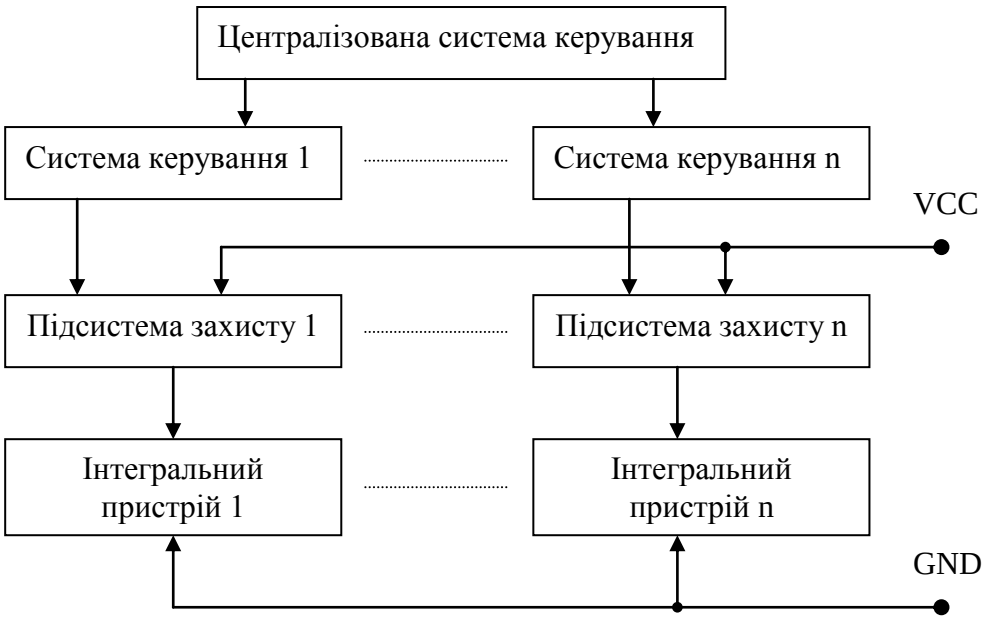


Рис.1.25. Структурна схема розподіленої системи захисту.

Робота окремих складових узгоджується відповідно до сигналів з централізованої системи керування. Внутрішні інтегральні пристрої, які треба захистити, такі як: центральний процесор та пам'ять даних живляться від окремих підсистем захисту. Модульна структура дозволяє розподіляти ресурси системи захисту у відповідності до виконуваної програми, щоб максимізувати рівень захищеності і мінімізувати витоки інформації. Інтегральне виконання такої структури та алгоритм для її системи керування є значно ускладнює систему.

Основні недоліки існуючих систем захисту:

1) Наведені системи захисту вирішують проблему несанкціонованого зчитування струму споживання лише частково – або зменшуючи корисний сигнал струму споживання, або вносячи додатковий шум до струму споживання, або використовуючи непрямі методи живлення мікроконтролера.

2) Алгоритм роботи систем є незмінним в процесі роботи. Зміна ПЗ часто є важливим, оскільки виробник не має змоги змінити апаратну частину свого пристрою у користувача, однак може ініціювати оновлення прошивки через мережу Інтернет.

3) Незалежно від реального струму споживання мікрокотнролера, системи захисту забезпечують завади тільки в заздалегідь визначеному діапазоні частот. Такі системи захисту є вразливими для диференційної атаки за струмом споживання (DPA).

1.6. Принципи побудови регульованих фільтрів для мікроконтролерів

Для створення системи захисту, яка б не мала наведених недоліків, або хоча б мінімізувала їх, слід визначити основні вимоги, до яких відносяться.

1) Алгоритм захисту системи має бути змінюваним без зміни апаратної частини, способом зміни прошивки мікроконтролера, з метою забезпечення можливості його вдосконалення.

2) Система має відслідковувати реальний струм споживання мікроконтролера, та на основі цих даних генерувати хибний струм споживання. Введення такого зворотного зв’язку дозволило б значно покращити захищеність від зчитування струму споживання.

Сучасні системи живлення мікроконтролерів. Традиційна система живлення мікроконтролера [26] складається із зовнішнього джерела живлення та фільтру, який живить безпосередньо центральний процесор та пам'ять у мікроконтролері (рис.1.26).

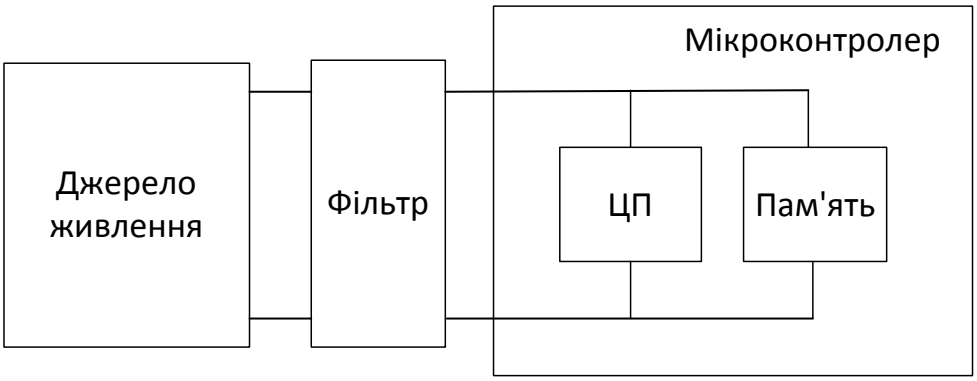


Рис.1.26. Структурна схема живлення мікроконтролера

Для того, щоб отримати інформацію про струм споживання мікроконтролера, зловмиснику слід підключитися між фільтром живлення та безпосередньо самим мікроконтролером. Таким чином, є можливість покращити захищеність мікропроцесорної системи від зчитування струму споживання шляхом переносу фільтру живлення всередину корпусу самого мікроконтролера, або в захищений корпус разом з мікроконтролером (рис 1.27).

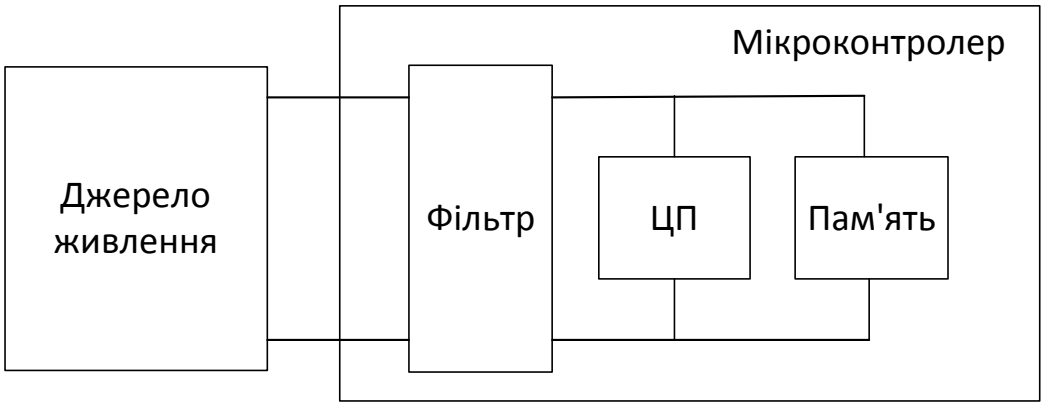


Рис.1.27. Перенесення фільтру живлення в мікроконтролер

Джерело живлення переносити в мікроконтролер не представляється можливим, через великі габарити джерела живлення та наявність елементів, які не можливо виконати інтегрально (трансформатори, дроселі). Крім того, смарт-карти живляться від зчитуючого пристрою, і не мають окремого джерела живлення.

Саме тому розробка фільтру живлення є найбільш перспективним з точки зору забезпечення захищеності від несанкціонованого зчитування за струмом споживання.

Регульовані фільтри для захисту інформації. Перспективним напрямком розвитку джерел живлення для мікропроцесорних систем є джерела живлення, що керуються за допомогою інформаційної шини зв'язку (рис.1.28). Введення інформаційної шини дає можливість створювати гнучкі та масштабовані системи електроживлення у сучасних цифрових пристроях на мікропроцесорах [27]. За допомогою інформаційної шини також стає можливим реалізувати алгоритми керування енергоспоживанням в пристроях, що живляться як від акумуляторів так і від мережі [28]. Наприклад, алгоритм керування варіює напругу живлення ядра мікропроцесора в залежності від заданого режиму роботи – енергозберігаючого режиму при роботі від акумуляторної батареї або режиму максимальної потужності при роботі від мережі.

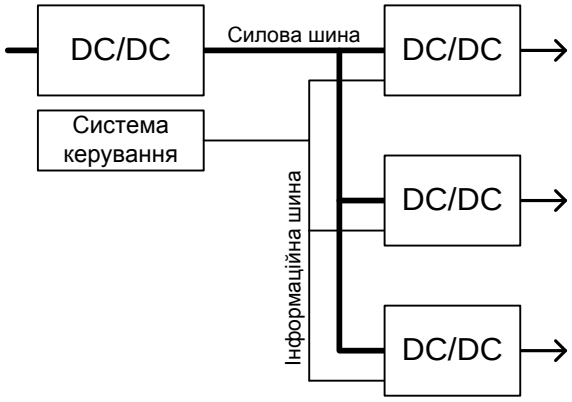


Рис.1.28. Система електроживлення з інформаційною шиною

За аналогією з керуванням перетворювачами за допомогою інформаційної шини, пропонується використання інформаційної шини для керування фільтром живлення (рис.1.29). Параметри фільтра є регульованими і задаються за допомогою системи керування (СК). Вимірювання струму споживання безпосереднього захищуваного центрального процесора ЦП та пам'яті

здійснюється за допомогою датчика струму та АЦП. Введення додаткового регулювання фільтром є оправданим, оскільки це дозволяє створити гнучкий алгоритм керування, який змінюється в процесі роботи. При цьому фільтр має складатися не тільки з ємностей та індуктивностей, як традиційний згладжуючий фільтр, але й містити додаткові елементи, які вносять шум до струму споживання.

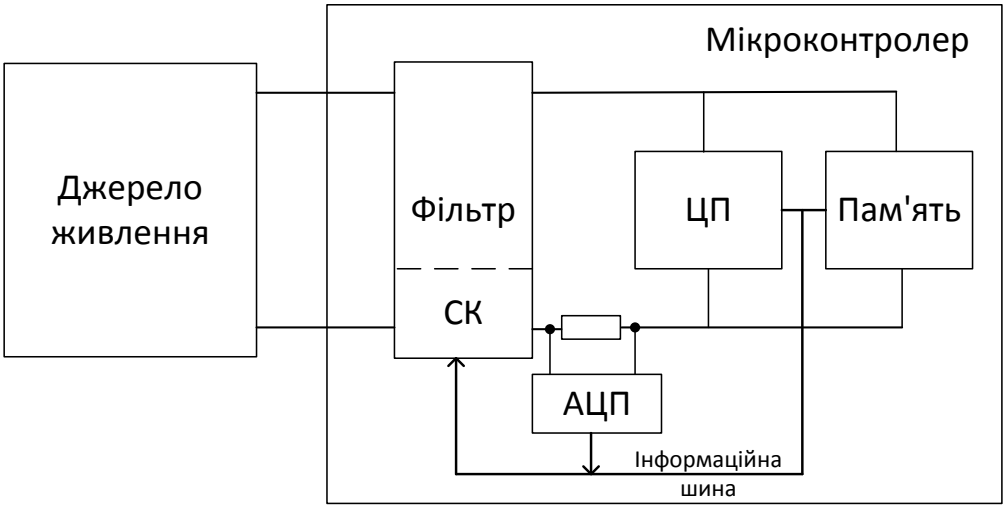


Рис.1.29. Фільтр з інформаційною шиною

Розглянуті структурні схеми фільтрів електроживлення можуть бути покладені в основу розробки систем захисту мікроконтролера за струмом споживання.

РОЗДІЛ 2

МАТЕМАТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ СТРУМУ СПОЖИВАННЯ У ПОЛЯРНИХ КООРДИНАТАХ

2.1. Методи аналізу дискретних сигналів струму споживання

Струм споживання мікроконтролера можна представити як сукупність дискретних значень на інтервалі, що дорівнює часу виконання команди.

Крім наведених вище способів аналізу струму споживання SPA і DPA у часовій області розглянемо аналіз дискретних функцій струму у спектральних областях та вейлвет-областях. Ці перетворення характеризуються меншою трудомісткістю при обчисленні, меншою кількістю базисних функцій. Саме ці особливості дозволяють реалізувати наведені перетворення за допомогою мікропроцесорної системи з обробкою даних у реальному масштабі часу.

Спектральний аналіз

Серед відомих способів спектральних перетворень дискретних функцій на кінцевих інтервалах є дискретне перетворення Фур'є, Хартлі [29], перетворення на скінченних інтервалах (СКІ-перетворення) [30] [31] та перетворення в орієнтованому базисі (ОБ-перетворення) [32].

Перетворення на скінченних інтервалах (СКІ-перетворення). Для автоматичної цифрової обробки сигналу струму споживання мікрокотролера доцільним є використання симетричного перетворення на кінцевих інтервалах (СКІ-перетворення) [33], оскільки воно має наступні переваги:

1) однаковий вигляд прямого та зворотного перетворень;
2) мінімальна кількість значень базисних функцій;
3) оперування тільки з дійсними числами.

Вищенаведені переваги СКІ-перетворення дозволяють досягнути найбільшої ефективності алгоритмів цифрової обробки сигналів, при їх реалізації у системах на мікропроцесорах або мікроконтролерах.

Пряме СКІ перетворення визначається наступним виразом:

$$Y(v) = \frac{1}{N}\sum_{x=0}^{N-1} y(x) \cdot \varphi(v,x) \qquad (2.1)$$

де $\varphi(v,x)$ - базисні функції;

$$\varphi(v,x) = cas\left(\frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)}x^{(s)}\right) = \cos\left(\frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)}x^{(s)}\right) + \sin\left(\frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)}x^{(s)}\right) \qquad (2.2)$$

$y(x)$ - решітчаста функція-оригінал, задана на кінцевій множині точок *N*.

Умовою, що накладається на решітчасту функцію, що підлягає СКІ перетворенню, є кінцевість її значень на інтервалі визначення *N*;

$Y(v)$ - зображення функції $y(x)$, що являє собою послідовність *N* дискретних значень; $v^{(s)}x^{(s)}$ - розрядні компоненти в *m*-ічному представленні чисел *x* та *v*.

Зворотне СКІ-перетворення, записане в скалярній формі, має вигляд:

$$y(x) = \sum_{v=0}^{N-1} Y(v) \cdot \varphi(v,x) \qquad (2.3)$$

Перетворення в орієнтованому базисі (СКІ-ОБ-перетворення) оперує з дискретними функціями (оригіналами і зображеннями), заданими на кінцевих інтервалах довжиною $N=m^n$, де *m,n* - цілі додатні числа, причому *m* – просте число [34] [35].

Перетворення має малу кількістю різних значень базисних функцій. В окремому випадку при *M* = 3 [36] базисні функції СКІ-ОБ-перетворення отримують цілі значення [37], що особливо важливо при реалізації алгоритмів обчислення спектрів та оригіналів на цілочисельних процесорах і мікроконтролерах.

Пряме перетворення в орієнтованому базисі має вигляд:

$$Y(\nu)=\sum_{x=0}^{N-1} y(x)\cdot\varphi_d(\nu,x), \qquad (2.4)$$

де $x,\ \nu = 0, 1,..., N-1$;

$$\varphi_d(\nu,x)=\cos\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right]+A\sin\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right] \qquad (2.5)$$

- базисні функції прямого перетворення; $y(x)$ - дискретна функція-оригінал; $A=tg\,\alpha$; $\alpha=\frac{2\pi i}{m}, i=\overline{1,m-1}$ – кут орієнтації вісі перетворення; $x^{(s)}, \nu^{(s)}$ - розрядні компоненти в m-ічному представленні чисел x та ν.

Зворотне перетворення вибирається з умови ортогональності матриць прямого і зворотного перетворення і описується виразом:

$$y(x)=\frac{1}{N}\sum_{\nu=0}^{N-1} Y(\nu)\cdot\varphi_r(\nu,x), \qquad (2.6)$$

де $\varphi_r(\nu,x)=\cos\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right]+\frac{1}{A}\sin\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right]$ (2.7)

- базисні функції зворотного перетворення.

При $A = 1$ функції прямого перетворення збігаються з функціями зворотного перетворення, а саме перетворення збігається з симетричним перетворенням на кінцевих інтервалах (СКІ-перетворенням).

Перетворення в орієнтованому базисі має можливість адаптації до конкретних способів його використання. Ця адаптація полягає у зміні кута орієнтації осі ОБ-перетворення. Враховуючи, що зміна призводить до m-1 значенню кута, а формули прямого і зворотного перетворень можуть мінятися місцями, кількість можливих модифікованих перетворень становить 2(m-1). Залежно від типу завдання, форми сигналів, які обробляються, користуються тим чи іншим варіантом перетворення, що забезпечує більшу гнучкість при побудові алгоритмів керування

За швидкодією перетворення в орієнтованому базисі не поступається перетворенню Уолша [34], проте на відміну від нього може оперувати з

інтервалом, кратним m (m – просте число). При m= 3 базисні функції приймають значення 0, 1, -1 (для прямого перетворення) або 0, 1, -2 (для зворотного). Обчислювальні операції лише з цілими числами також підвищують точність і швидкодію обробки даних. Ця обставина дозволяє ефективно використати перетворення в різноманітних системах обробки сигналів, наприклад, в системах зв'язку, що містять узгоджені фільтри.

Поряд із спектральними методами для дослідження спектральних функцій широко використовують вейвлет- аналіз [52,53]. Вейвлет- аналіз на базі функцій Хаара та ОБ наведено у Додатку А.

Наступним кроком аналізу періодичних функцій є використання полярних координат.

2.2. Представлення струму споживання в полярних координатах

При вимірюванні струму споживання за допомогою АЦП, струм представляється у вигляді відліків, розташованих через рівні проміжки часу (період дискретизації). Струм споживання являє собою періодичну функцію, оскільки час виконання команд є постійною величиною. Значення струму на початку періоду дорівнює значенню в кінці періоду. З огляду на наведені особливості струму споживання мікроконтролера, доцільно його представлення в полярній системі координат, оскільки така система координат дає наглядне представлення періодичних даних у вигляді замкнених фігур.

Існує певний клас функцій, подання яких у полярних координатах є більш простим і компактним, ніж подання цих функцій у декартових координатах. Зокрема, такими є функції, що описують окружність, еліпс та тригонометричні функції. Крім того, дані перетворення мають загальне значення при вирішенні прикладних задач.

Полярна система координат на площині задається точкою O (полюс) та направленою полярною віссю Ox. З кожною точкою P площини, на якій задана полярна система координат, можна співставити пару чисел ρ,φ (полярні

координати). На відміну від декартової системи координат, полярна система координат встановлює відповідність між парами чисел (ρ,φ), та точками на площині з точністю до $2\pi n+\varphi$.

При дослідженні струму споживання мікроконтролера, вимірювання проводять на певному часовому пероді. Так, при DPA-аналізі необхідно виділити період, на якому струм споживання повторюється, та обчислювати різницю між середнім струмом споживання, та струмом споживання при виконанні крипографічних операцій. Використання цифрового осцилографа або АЦП з зовньшньою синхронізацюєю також часто вимагає наявності синхронізації та вимірювання струму на окремому періоді повторення. В тому випадку, коли струм споживання повторюється з деяким періодом $t\in[0;T]$, можна скласти функцію струму споживання у полярних координатах, при цьому $\varphi\in[0;2\pi]$. Для переходу з декартової системи координат в полярну використовуються співвідношення [38]

$$\varphi=\frac{t\cdot 2\pi}{T},\ \rho=y \tag{2.8}$$

Для представлення струму споживання в полярних координатах отримаємо із вихідної послідовності АЦП послідовність дискретних точок у декартових координатах [39] (рис.2.1.а) та за допомогю співвідношень (2.8) перейдемо до полярних координат (рис.2.1.б).

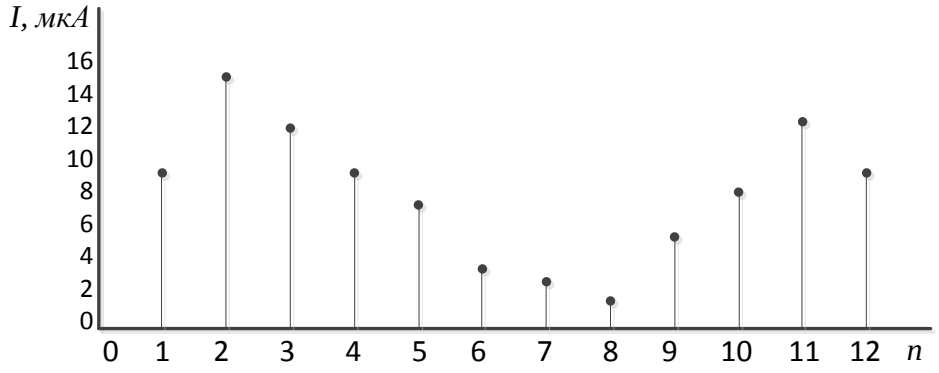


Рис.2.1. а) Подання струму споживання в декартовій системі координат,

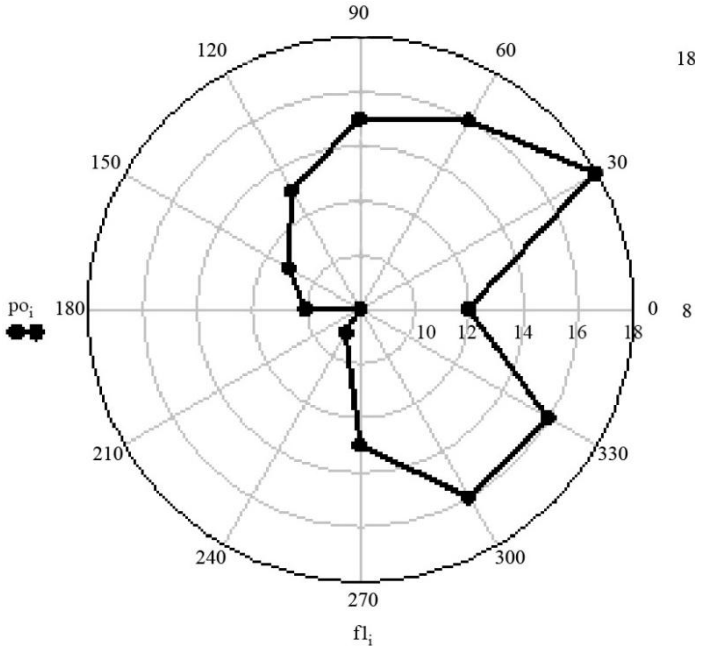


Рис.2.1. б) Подання струму споживання в полярній системі координат

В тому випадку, коли необхідно збереження форми кривої при переході між декартовою та полярною системою координат, при умові, що для вимірювання ρ, x, та y використані ті ж самі одиниці масштабу, декартові та полярні системи координат можуть бути пов'язані наступними співвідношеннями:

$$\begin{cases} x = \rho\cos\varphi \\ y = \rho\sin\varphi \end{cases} \tag{2.9}$$

$$\begin{cases} \rho = \sqrt{x^2 + y^2} \\ tg\varphi = \dfrac{y}{x} \end{cases} \tag{2.10}$$

де (x, y) - координати точки в декартовій системі координат, (ρ, φ) - координати точки в полярній системі координат.

Інтерполяція даних в полярних координатах

При вимірюванні струму споживання мікроконтролера, дані, що знаходятся між відліками АЦП, є невідомими для системи обробки інформації. Чим більша швидкість роботи АЦП, тим більше відліків за один і той самий

момент часу можна отримати, однак частота перетворення АЦП є величиною обмеженою, а високошвидкісні АЦП мають високу вартість. Для того, щоб отримати значення сигналу у моменти часу між відліками АЦП, використовують інтерполяцію. Найбільш поширеними методами інтерполяції є лінійна та сплайнова інтерполяції [40] [38]. При сплайновій інтерполяції використовуються поліноми другого та третього ступеня. Сплайнова інтерполяція є ефективною для функцій, які досить швидко змінюються, і не мають розривів в похідних.

На рис.2.2 точками показані відліки АЦП, та їх лінійна (Linear Interpolation) та сплайнова (Spline Interpolation) інтерполяції.

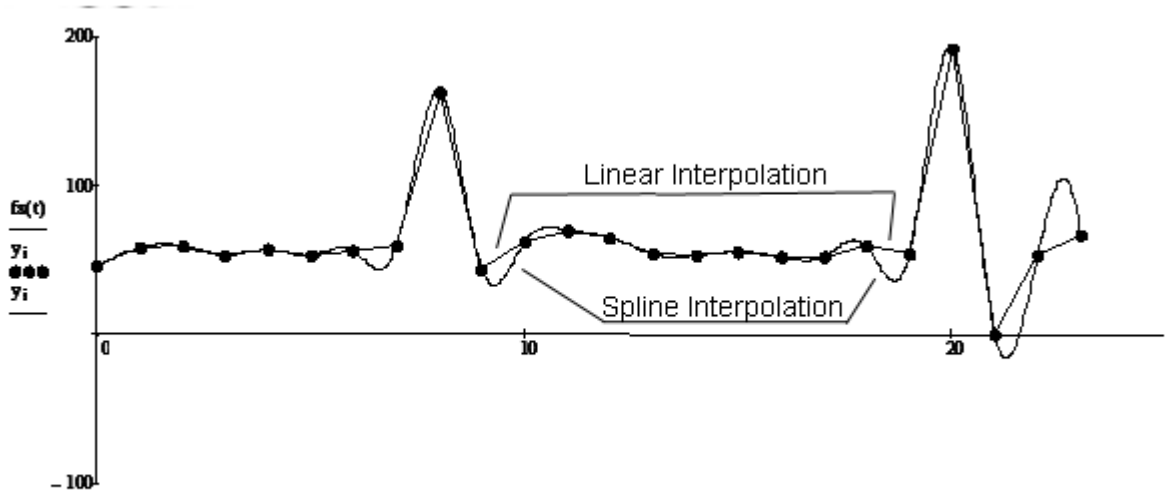


Рис.2.2. Відліки АЦП та їх сплайнова та лінійна інтерполяції в декартових координатах

В подальшому будемо використовувати лінійну інтерполяцію в полярних координатах. Вхідими даними для інтерполяції в полярних координатах є послідовність N дискретних точок $(p_0,\varphi_0),(p_1,\varphi_1),...,(p_i,\varphi_i),(p_{i+1},\varphi_{i+1}),...,(p_{N-1},\varphi_{N-1})$, що з'єднуються між собою відрізками прямих (рис.2.1).

Розглянемо детально отримання виразу інтерполюючої прямої між двома сусідніми точками (p_i,φ_i) та (p_{i+1},φ_{i+1}) (рис.2.3).

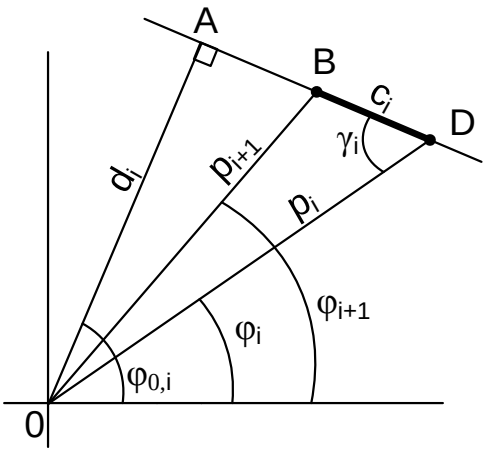


Рис.2.3. Визначення інтерполюючої прямої

Як відомо [41] [38], рівняння прямої в полярних координатах задається у вигляді:

$$p(\varphi) = \frac{d_i}{\cos(\varphi - \varphi_{0,i})} \tag{2.11}$$

де d_i - довжина перпендикуляру, опущеного з прямої, що відповідає і-му апроксимуючому відрізку до точки O, $\varphi_{0,i}$ - кут між перпендикуляром та полярною віссю.

Знайдемо значення кута $\varphi_{0,i}$ з прямокутного трикутника ADO:

$$\varphi_{0,i} = \frac{\pi}{2} - \gamma_i + \varphi_i \tag{2.12}$$

Знайдемо значення кута γ_i за теоремою косинусів:

$$\cos\gamma_i = \frac{c_i^2 + p_i^2 - p_{i+1}^2}{2c_i p_i} \tag{2.13}$$

$$\gamma_i = \arccos\left(\frac{c_i^2 + p_i^2 - p_{i+1}^2}{2c_i p_i}\right) \tag{2.14}$$

За теоремою косинусів знайдемо довжину відрізка c_i:

$$c_i = \sqrt{p_i^2 + p_{i+1}^2 - 2p_i p_{i+1}\cos(\varphi_i - \varphi_{i+1})} \tag{2.15}$$

З трикутника ADO, отримаємо довжину відрізка $d_i = p_i \sin\gamma_i$

Запишемо рівняння прямої на відрізку між точками (p_i, φ_i) та (p_{i+1}, φ_{i+1}):

$$p(\varphi) = \frac{p_i \sin\left(\arccos\left(\frac{2p_i - 2p_{i+1}\cos(\varphi_i - \varphi_{i+1})}{2\sqrt{p_i^2 + p_{i+1}^2 - 2p_i p_{i+1}\cos(\varphi_i - \varphi_{i+1})}}\right)\right)}{\cos\left(\varphi - \varphi_i - \frac{\pi}{2} + \arccos\left(\frac{2p_i - 2p_{i+1}\cos(\varphi_i - \varphi_{i+1})}{2\sqrt{p_i^2 + p_{i+1}^2 - 2p_i p_{i+1}\cos(\varphi_i - \varphi_{i+1})}}\right)\right)} \quad (2.16)$$

У результаті отримано спосіб інтерполяції для даних, заданих у полярних координатах. На рис.2.4 показано результат запропонованої інтерполяції у полярних координатах.

На рис.2.5 зображена лінійна інтерполяція в полярних координатах після зворотного переходу до декартових координат. Видно, що запропонований спосіб інтерполяції дає більш плавну форму кривих та має простий аналітичниій вираз для обчислення на кожному інтервалі у порівнянні зі сплайновою інтерполяцією.

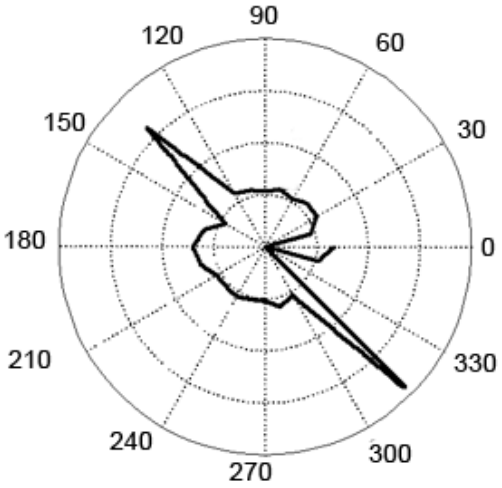


Рис.2.4. Лінійна інтерполяція в полярних координатах

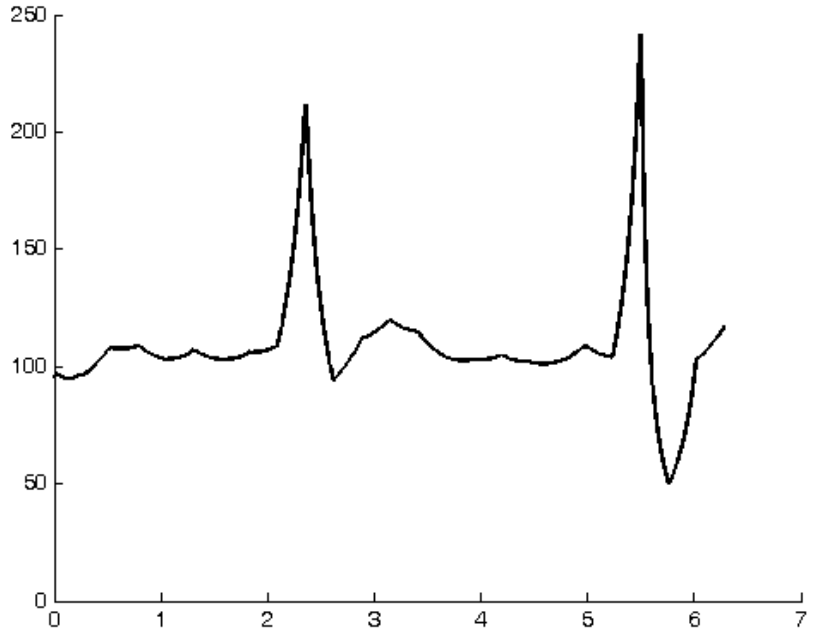


Рис.2.5. Лінійна інтерполяція в полярних координатах після зворотного переходу до декартових координат

2.3. Інтегральний показник струму споживання в полярних координатах

Геометрична інтерпретація перетворення на скінченних інтервалах (СКІ- перетворення) в полярних координатах

Розглянемо геометричну інтерпретацію прямого та зворотного СКІ перетворень, для чого будемо використовувати полярну систему координат, як найбільш адаптовану для представлення тригонометричних функцій.

Площа трикутника (рис.2.6), що утворений точками *O*, (ρ_1,φ_1) та (ρ_2,φ_2), обчислюється за формулою

$$S=\frac{1}{2}\begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix}=\frac{1}{2}\left(x_1y_2-x_2y_1\right) \qquad (2.17)$$

у декартових координатах, або за формулою:

$$S=\frac{1}{2}\begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix}=\frac{1}{2}\left(x_1y_2-x_2y_1\right) \qquad (2.18)$$

у полярних координатах.

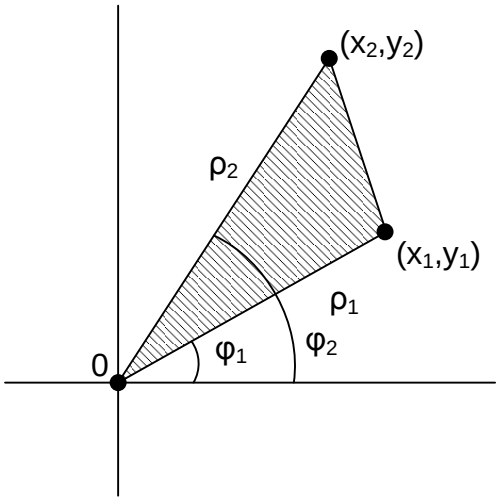


Рис. 2.6. Трикутник в полярних координатах.

Використовуючи властивості тригонометричних функцій, перетворимо вираз (2.18) наступним чином:

$$S = \frac{1}{2}\rho_1\rho_2(\sin\varphi_2\cos\varphi_1 - \cos\varphi_2\sin\varphi_1) = \frac{1}{2}\rho_1\rho_2\cos\varphi_1(\sin\varphi_2 - tg\varphi_1\cos\varphi_2) \qquad (2.19)$$

Для приведення виразу (2.19) до вигляду базисної функції СКІ перетворення, задаємо значення довжини радіус-векторів $\rho_1 = \frac{4}{\sqrt{2}}$, $\rho_2 = 1$, а значення кута $\varphi_1 = -\frac{\pi}{4}$. При цьому $tg\left(-\frac{\pi}{4}\right) = -1$, $\cos\left(-\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$.

Підставивши обчислені значення у вираз (2.19), отримаємо:

$$S = \frac{1}{2}\frac{\sqrt{2}}{2}(\sin\varphi_2 + \cos\varphi_2) = \frac{\sqrt{2}}{4}(\sin\varphi_2 + \cos\varphi_2) \qquad (2.20)$$

Вираз (2.20) визначає площу трикутника, утвореного двома радіус-векторами, полярний кут першого радіус-вектора фіксований, і дорівнює

$\varphi_1 = -\frac{\pi}{4}$, а полярний кут другого радіус вектора задається як $\varphi_2 = \frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)} i^{(s)}$,

$v, i = 0,1,..., N-1$ (рис.2.7).

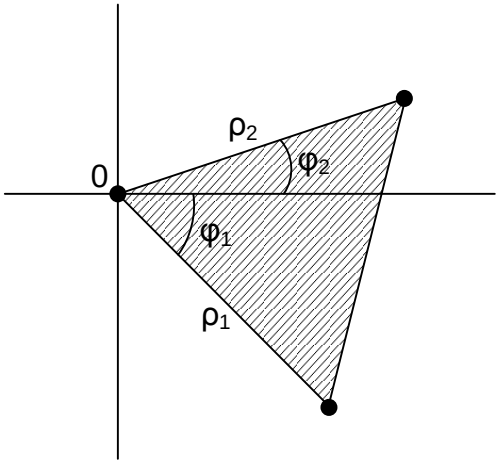


Рис.2.7. Трикутник в полярних координатах при $\rho_1 = \frac{4}{\sqrt{2}}$, $\rho_2 = 1$, $\varphi_1 = -\frac{\pi}{4}$

Підставивши у вираз (2.20) значення кута φ_2, отримаємо наступний вираз для базисної функції СКІ перетворення:

$$\varphi_{CKI}^{(P)}(v,\phi) = \sin\left(\frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)}\phi^{(s)}\right) + \cos\left(\frac{2\pi}{m}\sum_{s=1}^{n} v^{(s)}\phi^{(s)}\right) \qquad (2.21)$$

Таким чином, вираз площі трикутника на рис 2.6 співпадає з базисними функціями СКІ перетворення (2.21). Пряме СКІ перетворення в полярних координатах задається наступним виразом:

$$Y(v) = \frac{1}{N}\cdot\sum_{i=0}^{N-1}\rho(\phi_i)\cdot\varphi_{CKI}^{(P)}(v,\phi) \qquad (2.22)$$

Відповідно, зворотне СКІ перетворення в полярних координатах має вигляд:

$$\rho(\phi)=\sum_{v=0}^{N-1}Y(v)\cdot\left(\sin\left(\frac{2\pi}{m}\sum_{s=1}^{n}v^{(s)}\phi^{(s)}\right)+\cos\left(\frac{2\pi}{m}\sum_{s=1}^{n}v^{(s)}\phi^{(s)}\right)\right) \qquad (2.23)$$

Вищенаведені вирази відповідають виразам для СКІ-перетворення в декартових координатах.

Перетворення в орієнтованому базисі (ОБ-перетворення) в полярних координатах

Геометричну інтерпретацію базисних функцій ОБ-перетворення [42] отримаємо за аналогією з геометричною інтерпретацією СКІ-перетворення. Покладемо $\varphi_1=-\alpha$ (α – кут орієнтації осі перетворення) $\rho_1=\frac{1}{\cos\varphi_1}$, $\rho_2=1$.

Тоді вираз (2.20) визначає площу трикутника, утвореного двома векторами ρ_1 і ρ_2, при чому $\varphi_1=-\alpha$, а $\varphi_2=\phi_i\sum_{s=1}^{n}v^{(s)}i^{(s)}$.

Базисні функції прямого ОБ-перетворення в полярних координатах визначаються як:

$$\phi_0\sum_{s=1}^{n}v^{(s)}i^{(s)} \qquad \phi_0\sum_{s=1}^{n}v^{(s)}i^{(s)} \qquad (2.24)$$

—

$A=tg\,\alpha$.

$\alpha=\frac{2\pi k}{N}, k=\overline{1,N-1}$ – кут орієнтації вісі перетворення.

Базисні функції зворотного ОБ-перетворення

$$\phi_0\sum_{s=1}^{n}v^{(s)}i^{(s)} \quad - \quad \phi_0\sum_{s=1}^{n}v^{(s)}i^{(s)} \qquad (2.25)$$

Орієнтоване пряме ОБ-перетворення в полярних координатах ставить у відповідність до дискретної полярної функції зображення :

$$Y^{(p)}(v) = \frac{1}{N}\sum_{i=0}^{N-1}\rho(\phi_i)\varphi_d^{(p)}(v,\phi_i). \qquad (2.26)$$

Перехід в область оригіналів відбувається за формулою:

$$p(\phi_i) = \sum_{v=0}^{N-1} Y^{(p)}(v)\varphi_r^{(p)}(v,\phi_i). \qquad (2.27)$$

Для ОБ-перетворення в полярних координатах справедливі всі ті ж властивості базисних функцій у часовій області та теореми спектрального аналізу, що і для звичайного ОБ-перетворення.

Вейвлети в полярних координатах

Оскільки струм споживання мікроконтролера представлений в полярних координатах, пропонується розробити вейвлет-перетворення в полярних координатах для ідентифікації та виявлення трендів та деталізації [43] [44].

Вейвлет-перетворення Хаара в полярних координатах. За аналогією з перетворенням Хаара (див. Додаток А) побудуємо вейвлет-перетворення в полярній системі координат. Задамо область визначення аргументів вейвлетів як $\alpha = [0...2\pi)$. Тоді вирази для вейвлетів у полярній системі координат (далі полярних вейвлетів) приймуть вид:

$$\varphi^{(p)}{}_{0,0}(\alpha) = \begin{cases} 1, & 0 \le \alpha < 2\pi \\ 0, & \alpha < 0, \alpha \ge 2\pi \end{cases} \qquad (2.28)$$

$$\psi^{(p)}{}_{0,0}(\alpha)=\begin{cases}1, & 0\le\alpha<\pi\\ -1, & \pi\le\alpha<2\pi\\ 0, & \alpha<0,\alpha\ge2\pi\end{cases} \tag{2.29}$$

Функція $\varphi^{(p)}{}_{0,0}(\alpha)$ являє собою коло одиничного радіуса, а функція $\psi^{(p)}{}_{0,0}(\alpha)$ - півколо, причому зміна знака радіуса-вектора ρ в точці π виражається поворотом функції на кут π. Графічне подання полярних вейвлетів представлене на рис.2.8.

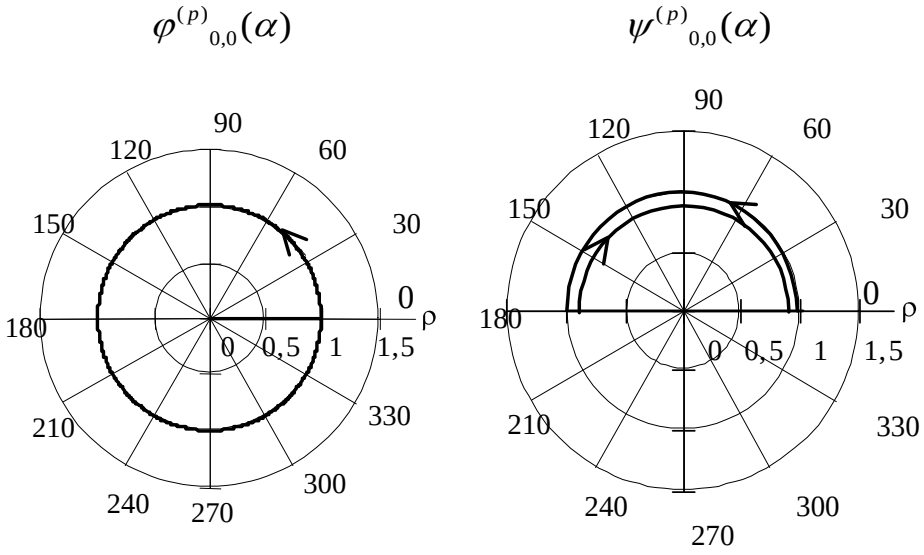


Рис.2.8. Функції $\varphi^{(p)}{}_{0,0}(x)$ та $\psi^{(p)}{}_{0,0}(x)$

Повний базис функцій полярного вейвлет-перетворення Хаара формується масштабованими (множення аргументу на 2^j, де j -рівень розкладання) і зміщеними (на величину k) скейлинг-функцією і материнським вейвлетом аналогічно звичайному перетворенню Хаара:

$$\begin{aligned}\varphi^{(p)}{}_{j,k}&=2^{j/2}\varphi^{(p)}(2^j\alpha-k)\\ \psi^{(p)}{}_{j,k}&=2^{j/2}\psi^{(p)}(2^j\alpha-k)\end{aligned} \tag{2.30}$$

Застосувавши рівняння (2.30) до рівнянь (2.28) і (2.29) одержимо запис полярних вейвлетів Хаара $\varphi^{(p)}{}_{j,k}(x),\psi^{(p)}{}_{j,k}(x)$, для будь-якого рівня розкладання:

$$\varphi_{j,k}(\alpha)=\begin{cases}2^{j/2}, & \dfrac{2\pi k}{2^j}\le\alpha<\dfrac{2\pi(k+1)}{2^j}\\ 0, & \alpha<\dfrac{2\pi k}{2^j},\alpha\ge\dfrac{2\pi(k+1)}{2^j}\end{cases} \tag{2.31}$$

$$\psi_{j,k}(\alpha)=\begin{cases}2^{j/2}, & \dfrac{4\pi k}{2^{j+1}}\le\alpha<\dfrac{2\pi(2k+1)}{2^{j+1}}\\ -2^{j/2}, & \dfrac{2\pi(2k+1)}{2^{j+1}}\le\alpha<\dfrac{4\pi(k+1)}{2^{j+1}}\\ 0, & \alpha<\dfrac{4\pi k}{2^{j+1}},\alpha\ge\dfrac{4\pi(k+1)}{2^{j+1}}\end{cases} \tag{2.32}$$

Пряме вейвлет-перетворення Хаара в полярних координатах записується наступним чином:

$$\begin{aligned}s_{j-1,k}&=\frac{1}{\sqrt{2}}\varphi(\alpha)\cdot \mathrm{S_j},\\ d_{j-1,k}&=\frac{1}{\sqrt{2}}\psi(\alpha)\cdot \mathrm{S_j}.\end{aligned} \tag{2.33}$$

де j - рівень розкладання $j=\overline{0,p}$, S_j – вектор-стовпчик коефіцієнтів апроксимації

$$\mathrm{S_j}=\begin{bmatrix}s_{j,2k} & s_{j,2k+1}\end{bmatrix}^T. \tag{2.34}$$

Для найменшого масштабу, котрий відповідає $j_{\max}$, розраховуються лише коефіцієнти апроксимації $s_{j_{\max},k}$

$$s_{j_{\max},k}=\frac{f(\alpha/K)}{2^{j_{\max}/2}}, \tag{2.35}$$

де f(α/K) – значення функції-оригіналу в точці α на інтервалі визначення $K=2^{j_{\max}}$.

Зворотне полярне перетворення записується через коефіцієнти розкладання (апроксимації і деталізації) як:

$$s_{j,2k} = \frac{1}{\sqrt{2}}\varphi(\alpha)\cdot \mathrm{D},$$
$$s_{j,2k+1} = \frac{1}{\sqrt{2}}\psi(\alpha)\cdot \mathrm{D}, \tag{2.36}$$

де D – вектор-стовпчик коефіцієнтів розкладання на (j-1)-му рівні розкладання:

$$\mathrm{D} = \left[s_{j-1,k} \;\; d_{j-1,k} \right]^T. \tag{2.37}$$

На найбільшому масштабі, який відповідає j=0 і співпадає з інтервалом визначення функції-оригінала K, функція-оригінал представляється через коефіцієнти розкладання наступним чином:

$$f(x) = s_{0,0}\varphi_{0,0}^{(p)}(\alpha) + d_{0,0}\psi_{0,0}^{(p)}(\alpha) +$$
$$+\sum_{k=0}^{1} d_{1,k}\psi_{1,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{2^j-1} d_{j,k}\psi_{j,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{2^{(p-1)}-1} d_{p-1,k}\psi_{j_{\max}-1,k}^{(p)}(\alpha). \tag{2.38}$$

Приклад 2.1. Нехай задана дискретна функція (рис.2.9), визначена на інтервалі $K = 2^{j_{\max}} = 16$, $j_{\max} = 4$. Необхідно знайти вейвлет-коефіцієнти й представити функцію на різних рівнях розкладання. Першим кроком вейвлет-аналізу є обчислення коефіцієнтів апроксимації $s_{j_{\max},k}$. Наступними кроками будуть обчислення вейвлет-коефіцієнтів. Реконструкція функції на рівнях $j = 3,2,1,0$ представлена в табл.2.1.

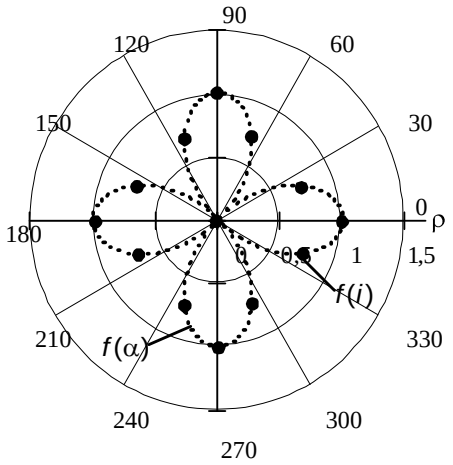
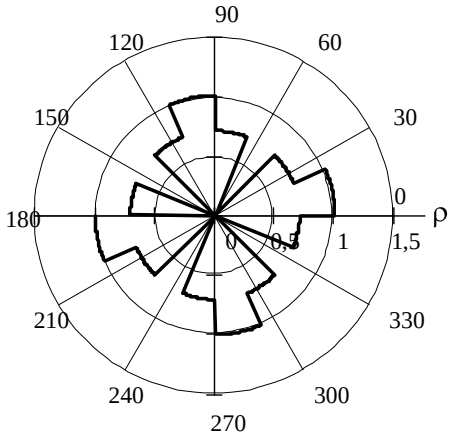


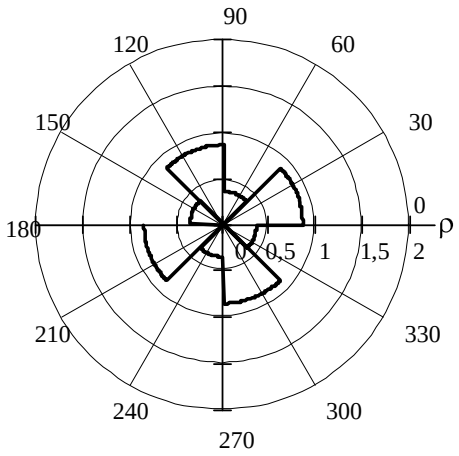
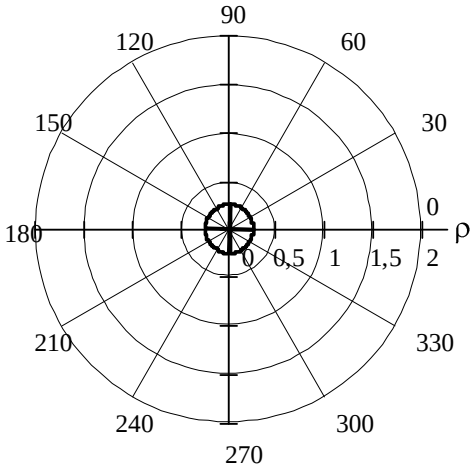
Рис.2.9. Неперервна функція $\rho(\varphi)$ (пунктир),
та її дискретний аналог $\rho(i)$ (точки).

Відновлена функція збігається з функцією для рівня розкладання при j=3, тобто функція відновлюється без втрат.

Таблиця 2.1

Реконструкція функції в полярних координатах на рівнях $j = 3,2,1,0$ після вейвлет-розкладу Хаара.

Графік функції	**Рівень розкладання**
	$j = 3$

	$j=2$
	$j=1$

Вейвлет-перетворення на базі ОБ-перетворення в полярних координатах. Для подання полярного ОБ вейвлет-перетворення запишемо базисні вейвлет-функції для прямого та зворотного полярного ОБ-вейвлет перетворення у вигляді функцій з областю визначення аргументу $\alpha=[0...2\pi)$:

$$\varphi^{(p)}{}_d(\alpha)=\varphi^{(p)}{}_r(\alpha)=\begin{cases}3^{j/2}, & \dfrac{2\pi k}{3^j}\le\alpha<\dfrac{2\pi(k+1)}{3^j}\\ 0, & \alpha<\dfrac{2\pi k}{3^j},\alpha\ge\dfrac{2\pi(k+1)}{3^j}\end{cases} \qquad (2.39)$$

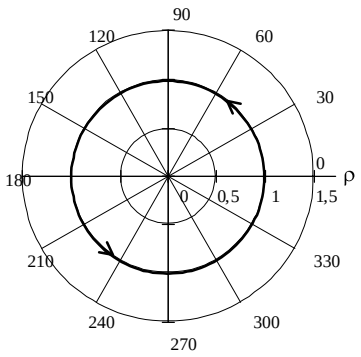
$$\psi_d(\alpha)=\begin{cases}3^{j/2}, & \frac{6\pi k}{3^{j+1}}\le\alpha<\frac{2\pi(3k+1)}{3^{j+1}}\\ -3^{j/2}, & \frac{2\pi(3k+1)}{3^{j+1}}\le\alpha<\frac{2\pi(3k+2)}{3^{j+1}}\\ 0, & \alpha<\frac{6\pi k}{3^{j+1}},\alpha\ge\frac{6\pi(k+1)}{3^{j+1}}\end{cases} \tag{2.40}$$

$$\gamma^{(p)}{}_d(\alpha)=\begin{cases}3^{j/2}, & \frac{6\pi k}{3^{j+1}}\le\alpha<\frac{2\pi(3k+1)}{3^{j+1}}\\ 0, & \frac{2\pi(3k+1)}{3^{j+1}}\le\alpha<\frac{6\pi(k+1)}{3^{j+1}}\\ -3^{j/2}, & \alpha<\frac{6\pi k}{3^{j+1}},\alpha\ge\frac{6\pi(k+1)}{3^{j+1}}\end{cases} \tag{2.41}$$

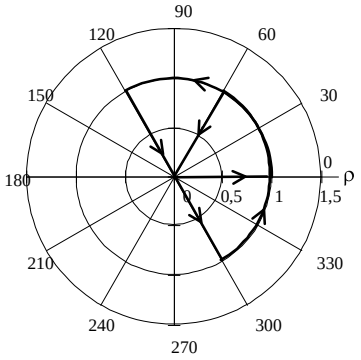
$$\psi^{(p)}{}_r(\alpha)=\begin{cases}3^{j/2}, & \frac{6\pi k}{3^{j+1}}\le\alpha<\frac{2\pi(3k+1)}{3^{j+1}}\\ -2\cdot3^{j/2}, & \frac{2\pi(3k+1)}{3^{j+1}}\le\alpha<\frac{2\pi(3k+2)}{3^{j+1}}\\ 3^{j/2}, & \frac{2\pi(3k+2)}{3^{j+1}}\le\alpha<\frac{6\pi(k+1)}{3^{j+1}}\\ 0, & \alpha<\frac{6\pi k}{3^{j+1}},\alpha\ge\frac{6\pi(k+1)}{3^{j+1}}\end{cases} \tag{2.42}$$

$$\gamma^{(p)}{}_r(\alpha)=\begin{cases}3^{j/2}, & \frac{6\pi k}{3^{j+1}}\le\alpha<\frac{2\pi(3k+1)}{3^{j+1}}\\ 3^{j/2}, & \frac{2\pi(3k+1)}{3^{j+1}}\le\alpha<\frac{2\pi(3k+2)}{3^{j+1}}\\ -2\cdot3^{j/2}, & \frac{2\pi(3k+2)}{3^{j+1}}\le\alpha<\frac{6\pi(k+1)}{3^{j+1}}\\ 0, & \alpha<\frac{6\pi k}{3^{j+1}},\alpha\ge\frac{6\pi(k+1)}{3^{j+1}}\end{cases} \tag{2.43}$$

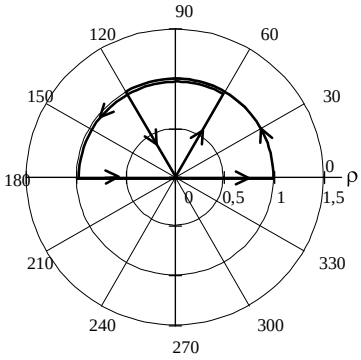
У полярній системі координат наведені функції (2.39) - (2.43) мають вигляд, представлений на рис.2.10. Стрілки на рис.2.10. вказують напрям зміни функції при зміні кута від 0 до 2π.



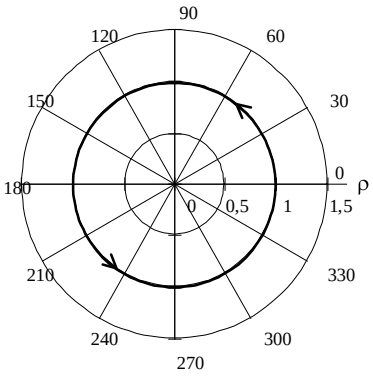
$$\varphi^{(p)}{}_{d}(\alpha)=\{1;1;1\}$$



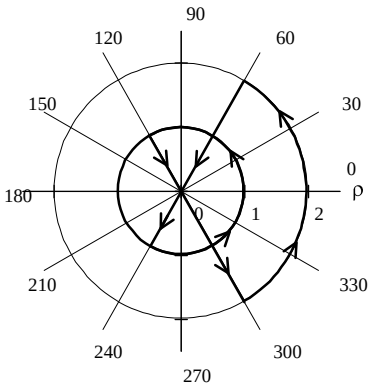
$$\psi^{(p)}{}_{d}(\alpha)=\{1;-1;0\}$$



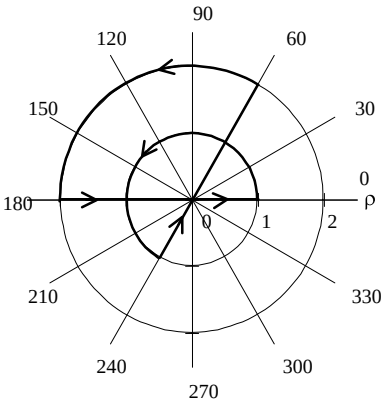
$$\gamma^{(p)}{}_{d}(\alpha)=\{1;0;-1\}$$



$$\varphi^{(p)}{}_{r}(\alpha)=\{1;1;1\}$$



$$\psi^{(p)}{}_{r}(\alpha)=\{1;-2;1\}$$



$$\gamma^{(p)}{}_{r}(\alpha)=\{1;1;-2\}$$

Рис.2.10. Базисні вейвлет-функції для прямого та зворотного полярного ОБ-вейвлет перетворення.

Визначення ОБ-вейвлет-спектру для функції, заданої в полярних координатах проводиться з використанням базисних функцій прямого вейвлет-пертворення $\varphi^{(p)}{}_{d}(\alpha),\psi^{(p)}{}_{d}(\alpha),\gamma^{(p)}{}_{d}(\alpha)$. Для відновлення вихідної функції із вейвлет-спектру використовуються базисні функції зворотного перетворення $\varphi^{(p)}{}_{r}(\alpha),\psi^{(p)}{}_{r}(\alpha),\gamma^{(p)}{}_{r}(\alpha)$.

Пряме ОБ вейвлет-перетворення в полярних координатах записується наступним чином:

$$s_{j-1,k} = \frac{1}{\sqrt{3}}\varphi_d^{(p)}(\alpha)\cdot \mathrm{S_j};$$
$$d_{j-1,k} = \frac{1}{\sqrt{3}}\psi_d^{(p)}(\alpha)\cdot \mathrm{S_j}; \qquad (2.44)$$
$$l_{j-1,k} = \frac{1}{\sqrt{3}}\gamma_d^{(p)}(\alpha)\cdot \mathrm{S_j},$$

де $j = \overline{0, j_{\max}}$, k – інтервал розгляду на даному рівні j, $k = \overline{0,3^j - 1}$;

$$\mathbf{S_j} = \left[s_{j,3k} \;\; s_{j,3k+1} \;\; s_{j,3k+2} \right]^T. \qquad (2.45)$$

Зворотне ОБ вейвлет перетворення в полярних координатах проводиться через коефіцієнти перетворення як:

$$s_{j,3k} = \frac{1}{\sqrt{3}}\varphi_r^{(p)}(\alpha)\cdot \mathrm{D};$$
$$s_{j,3k+1} = \frac{1}{\sqrt{3}}\psi_r^{(p)}(\alpha)\cdot \mathrm{D}; \qquad (2.46)$$
$$s_{j,3k+2} = \frac{1}{\sqrt{3}}\gamma_r^{(p)}(\alpha)\cdot \mathrm{D},$$

де D – вектор-стовпчик коефіцієнтів розкладання на (j-1)-му рівні розгляду:

$$\mathrm{D} = \left[s_{j-1,k} \;\; d_{j-1,k} \;\; l_{j-1,k} \right]^T. \qquad (2.47)$$

На найбільшому масштабі при j=0 функція-оригінал представляється наступним чином:

$$f(\alpha) = s_{0,0}\varphi_{0,0}^{(p)}(\alpha) +$$
$$+d_{0,0}\psi_{0,0}^{(p)}(\alpha) + \sum_{k=0}^{2} d_{1,k}\psi_{1,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{3^j-1} d_{j,k}\psi_{j,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{3^{(-1)}-1} d_{p-1,k}\psi_{j_{\max}-1,k}^{(p)}(\alpha) + \qquad (2.48)$$
$$+l_{0,0}\gamma_{0,0}^{(p)}(\alpha) + \sum_{k=0}^{3} l_{1,k}\gamma_{1,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{3^j-1} l_{j,k}\gamma_{j,k}^{(p)}(\alpha) + \ldots + \sum_{k=0}^{3^{(j_{\max}-1)}-1} l_{p-1,k}\gamma_{j_{\max}-1,k}^{(p)}(\alpha),$$

де $\psi^{(p)} = \psi_r^{(p)}$, $\gamma^{(p)} = \gamma_r^{(p)}$ – базисні функції зворотного ОБ-вейвлат перетворення.

Приклад 2.2. Нехай вихідна функція-оригінал $f(\alpha)$ визначена на $3^{J_{\max}} = 3^3 = 27$ відліках. (рис.2.11). Необхідно знайти апроксимацію функції на рівнях розкладання для j=2, 1, 0 за допомогою полярного вейвлет ОБ-перетворення.

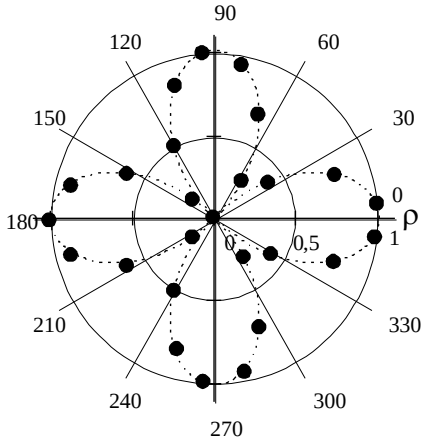


Рис.2.11. Дискретна функція

$$f(\alpha) = \sin^2 \alpha - \cos^2 \alpha .$$

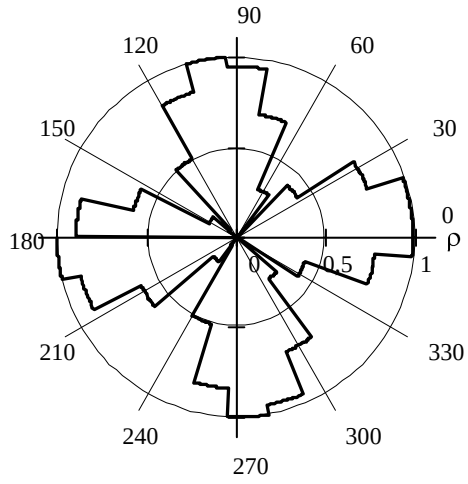
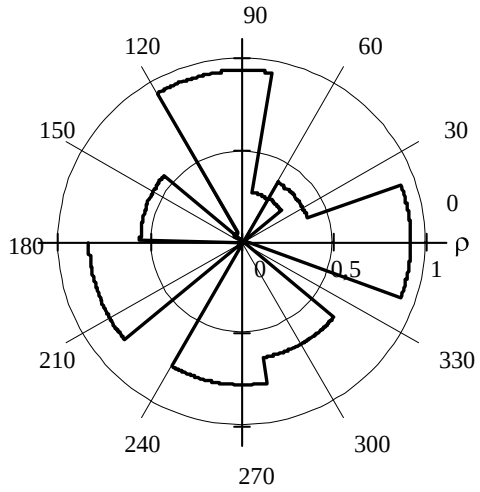
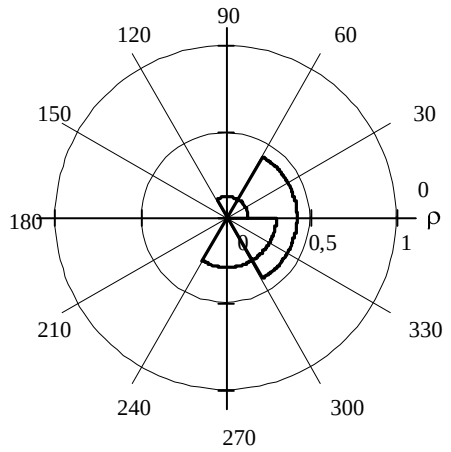
Отримані вейвлет-коефіцієнти $s_{j,k}; d_{j,k}; l_{j,k}$ по формулі (2.44) підставляємо послідовно для j=2,1,0. Результати розрахунку зведені в табл.2.2.

Відновлена функція має вигляд, аналогічний зазначеному виду в табл.2.2 для рівня розкладання j=2.

Функції $\varphi^{(p)}{}_{j,k}(\alpha)$, $\psi^{(p)}{}_{j,k}(\alpha)$ і $\gamma^{(p)}{}_{j,k}(\alpha)$ є аналогами ОБ-перетворення і описують кінцеву імпульсну характеристику "широкополосного" фільтра та двох "вузькополосних" фільтрів відповідно. Доданки з коефіцієнтами $d_{j,k}$ і $l_{j,k}$, де $j > 0$ вказують на флуктуації на менших масштабах з більшими j. У загальному випадку полярне ОБ-перетворення дозволяє одержати 3^j коефіцієнтів $s_{j,k}$ і по $3^{j_{\max}} - 3^j$ коефіцієнтів $d_{j,k}$ і $l_{j,k}$

Таблиця 2.2.

Вейвлет-розклад ОБ для функції в полярних координатах для рівнів розкладання j=2,1,0.

Графік функції	**Рівень розкладання**
	*j*=2
	*j*=1
	*j*=0

Представлення скейлинг-функції і материнський вейвлет ОБ вейвлет перетворення дозволяє зменшити кількість рівнів розкладання вейвлет аналізу для періодичних функцій.

2.4. Знаходження інтегрального показника струму споживання в полярних координатах

В якості критерію оцінки струму споживання пропонується використати площу багатокутника, що знаходиться під ломаною лінійної апроксимації в полярних координатах – інтегральний показник струму споживання в полярних координатах. Знайдемо інтегральний показник для вхідного масиву даних в полярних координатах. Для цього знайдемо площу трикутника, дві вершини якого знаходяться у точках φ_i та φ_{i+1}, а третя точка співпадає з початком координат:

$$S_i = \frac{1}{2} \cdot p_i \cdot p_{i+1} \cdot \sin\left(\varphi_{i+1} - \varphi_i\right) \tag{2.49}$$

У тому випадку, коли використовуються дані з АЦП, можна використати однаковий крок між сусідніми точками, оскільки частота дискретизації АЦП є фіксованою, і з часом, як правило, не змінюється.

Тому можна записати $\varphi_i = \frac{2\pi}{N} \cdot i$, та $\varphi_{i+1} = \frac{2\pi}{N} \cdot (i+1)$, після підстановки значення кутів у вираз (2.49) отримаємо:

$$S_i = \frac{1}{2} \cdot p_i \cdot p_{i+1} \cdot \sin\left(\frac{2\pi}{N} \cdot (i+1) - \frac{2\pi}{N} \cdot i\right) = \frac{1}{2} \cdot p_i \cdot p_{i+1} \cdot \sin\left(\frac{2\pi}{N}\right) \tag{2.50}$$

Додамо площі усіх трикутників, що містяться під кривою (див. рис 2.4):

$$S_\Sigma = \frac{1}{2} \cdot \sin\left(\frac{2\pi}{N}\right) \cdot \sum_{i=0}^{N-1} p_i \cdot p_{i+1} \tag{2.51}$$

Наведений вираз (2.51) дає інтегральний показник струму споживання, що характеризує енергію послідовності при поданні у полярних координатах. Чисельно вираз визначає площу фігури рис 2.1.б.

Інколи наближену інформацію про тип команди можна отримати на $\frac{1}{4}$, $\frac{1}{3}$ або $\frac{1}{2}$ інтервалу часу виконання команди на мікроконтролері.

Тоді має сенс користуватися інтегральними показниками типу:

$$\begin{aligned} S_{\frac{1}{4}} &= \frac{1}{2}\cdot\sin\left(\frac{2\pi}{N}\right)\cdot\sum_{i=0}^{N/4-1} p_i\cdot p_{i+1} \\ S_{\frac{1}{3}} &= \frac{1}{2}\cdot\sin\left(\frac{2\pi}{N}\right)\cdot\sum_{i=0}^{N/3-1} p_i\cdot p_{i+1} \\ S_{\frac{1}{2}} &= \frac{1}{2}\cdot\sin\left(\frac{2\pi}{N}\right)\cdot\sum_{i=0}^{N/2-1} p_i\cdot p_{i+1} \end{aligned} \qquad (2.52)$$

Вирази (2.52) дозволяють ідентифікувати команду з затримкою на $\frac{1}{4}$, $\frac{1}{3}$ або $\frac{1}{2}$ періоду і можуть бути використані в системах захисту із зворотним зв’язком з ідентифікацією поточної виконуваної команди.

Для кожної з команд вимірювалася інтегральний показник за формулою (2.51). Інтегральні показники, що характеризують середнє значення площі під кривою струму споживання даної команди у полярних координатах, зведені у табл.2.3. Кожна команда була відтворена 10 разів для покращення результатів вимірювань в цілому. Для зручності та наочності представлення даних, усі показники розділені на 1000 та округлені до найближчого цілого. Це дозволяє створити попереднє групування команд, використовуючи значення інтегрального показника.

Таблиця.2.3.

Групування команд за інтегральним показником струму споживання у полярних координатах

Команда	Інтегральний показник	Група	Команда	Інтегральний показник	Група
CPSE	3	1	SER	14	5
SUB	3	1	BRMI	8	5
COM	3	1	SEC	14	5
CLC	3	1	CLZ	9	6
SEZ	3	1	CLR	15	6
ROL	4	2	NEG	15	6
BRPL	4	2	CPI	16	7
NOP	4	2	CP(Z=1)	16	7
BRCC	6	3	BREQ	17	8
MOV	7	4	ROR	18	9
BRCS	14	5	BRNE	19	10
SWAP	14	5	CP(Z=0)	22	11
ADD	14	5	LDI	22	11

З табл.2.3 видно, що результати вимірювання струму споживання команд в цілому мають добру повторюваність інтегрального показника, та дана характеристика є різною для різних послідовностей команд. Групи команд розділяються достатньо чітко, всередині групи необхідно застосовувати додаткове дослідження для визначення відмінностей між струмами споживання. Однак для цілей організації захисту від зчитування за струмом споживання із зворотним зв’язком цього групування достатньо.

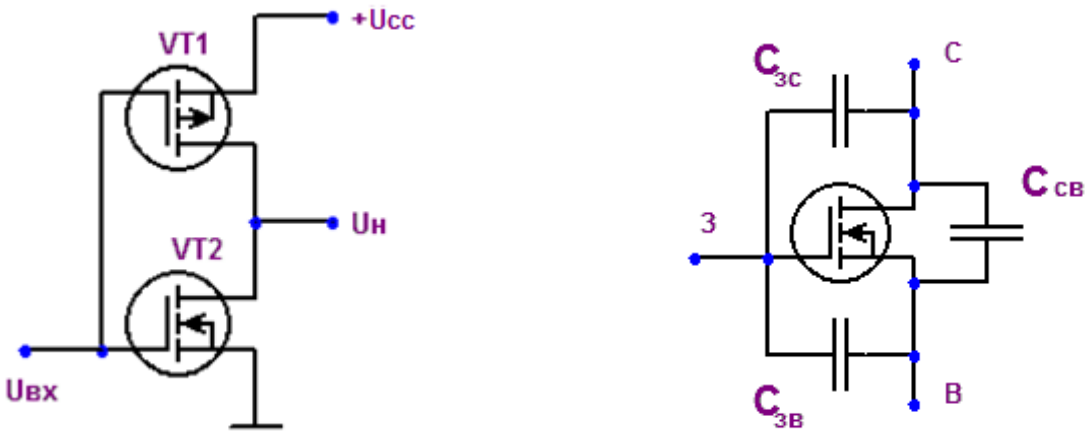
Таким чином, запропоновані вейвлет-перетворення в полярних координатах на основі перетворень Хаара та ОБ дозволяють отримувати більш прості вирази, та покращити наочність вейвлет-аналізу в тих випадках, коли функцію-оригінал доцільно представити в полярних координатах у порівнянні з представленням цієї ж функції в декартових координатах. Визначений зв'язок базисних функцій СКІ та значень площ сегментів функцій струму споживання в полярних координатах

Представлення струму в полярних координатах дозволило знайти вираз інтегрального показника струму, який характеризує енергію струму споживання при виконанні команди та легко обчислюється в реальному масштабі часу як площа під кривою лінійної апроксимації струму в полярних координатах. Інтегральний показник струму у полярних координатах має більш компактне подання та менший час обчислення у порівнянні з обчисленням коефіцієнта взаємної кореляції з усіма відомими послідовностями команд з бази даних.

ОЦІНКА СТУПЕНЮ ЗАХИЩЕНОСТІ МІКРОКОНТРОЛЕРІВ ВІД ЗЧИТУВАННЯ ЗА СТРУМОМ СПОЖИВАННЯ

3.1. Визначення внутрішнього стану мікроконтролера за струмом споживання

Розглянемо процес виникнення перехідних струмів у простих логічних схемах. Основний елемент такої логічної схеми – КМДН інвертор [45] [46] (рис.3.1.а). Інвертор побудовано на двох МДН транзисторах з ізольованим затвором з n та p каналами відповідно.



а) б)

Рис.3.1. а) Логічний інвертор, побудований за технологією КМДН на двох МДН транзисторах, б) МДН-транзистор та його основні паразитні ємнісні елементи.

При підключенні на вхід інвертора сигналу з рівнем логічної «1» відкривається транзистор VT2, а транзистор VT1 при цьому закрито. При відсутності навантаження струм споживання такого ланцюга буде практично дорівнювати нулю. Відсутність навантаження в даному випадку є штатним режимом роботи, оскільки інвертор всередині мікросхеми МК навантажено на затвор іншого логічного ланцюга. У випадку, коли на вхід розглядуваного

інвертора подано рівень логічного «0», відкривається транзистор VT1, а VT2 закривається низьким потенціалом затвор-виток. В цьому випадку струм споживання мікросхеми також досить малий. Враховуючи вищесказане, можна зробити висновок, що статичний струм споживання мікросхем та мікроконтролерів, побудованих за КМДН технологією не може буди вибраний у якості об'єкта для дослідження.

В динамічному режимі, коли відбувається переключення транзисторних МДН-ключів, виникають перехідні струми, які значно перевищують за амплітудою струми статичного режиму. При чому, чим більша частота перемикання транзисторів, тим більший вклад в сумарний струм споживання мікросхеми вноситимуть саме перехідні струми. Природа виникнення перехідних струмів пов'язана з наявністю паразитних ємнісних елементів у МДН транзисторі [26] [47] [48] [49] [50]. На рис 3.1.б показана модель МДН транзистора з урахуванням паразитних ємнісних елементів, де позначено: Сзв – ємність затвор-виток, Сзс – ємність затвор-сток, Ссв – ємність сток-виток. Величина паразитних ємностей МДН транзистора обмежує його максимальну робочу частоту, а також втрати на переключення. Саме зарядні струми вищезгаданих ємностей вноситимуть значний вклад в сумарний струм споживання досліджуваного мікроконтролера. Паразитні ємності транзистора тим більше, чим більше площа кристалу самого транзистора. Так, для потужних транзисторів значення ємності вище, ніж для малопотужних інтегральних транзисторів, які створюються у кристалі мікросхем.

Вимірювання струму споживання мікроконтролера проводиться за допомогою датчика струму, який виконується або як низькоомний резистор в ланцюзі навантаження, або як трансформатор струму. Використрання трансформатору струму в даному випадку дещо ускладнене тим, що наперед невідома максимальна частота гармонік струму навантаження та амплітуда струмового сигналу. Саме тому в роботі у якості датчика струму було використано низькоомний неіндуктивний резистор.

Вимірювання струму споживання повинно виконуватися швидкодійним цифровим осцилографом або АЦП зі смугою пропускання [51], щонайменше у 10 разів перевищуючою робочу частоту досліджуваного мікроконтролера. Розрядність АЦП повинна складати 8...16 біт для обробки сигналів з достатньою роздільною здатністю. Чим більшу розрядність має АЦП, тим точніше можна визначити відхилення струму споживання МК при виконанні тої чи іншої інструкції. За умови високої розрядності АЦП з'являється можливість виміряти невеликі флуктуації струму, спричинені обробкою даних в МК навіть для випадків, коли такі дані не передаються по внутрішній шині. Серед досліджуваних інструкцій мікроконтролера, особливу увагу викликають такі, яку оперують з регістром прапорців, особливо з прапорцем переносу, оскільки саме цей прапорець звичайно використовується в алгоритмах шифрування та дешифрування даних. Крім того, оскільки прапорці змінюють хід виконання програми, можна визначити прапорці шляхом аналізу послідовності команд, що виконуються.

Виконання інструкцій викликає різний рівень активності у декодері команд та у арифметично-логічному пристрої мікропроцесора, і за цією активністю можна відокремити одну послідовність команд від іншої та відновити певні частини алгоритму. Різні пристрої мікропроцесора дають відповідні перехідні процеси по відношенню до сигналів тактування, що дозволяє відокремити операції з цими пристроями у часі.

Струми споживання обов'язково мають шумові компоненти, які можуть потрапляти як з джерела живлення, так і наводитися зовнішніми полями. Вплив шумових компонентів можна знизити за допомогою:

1) фільтрації стаціонарних складових пасивними та активними згладжувальними фільтрами по ланцюгам живлення;

2) фільтрації нестаціонарних складових за допомогою фільтрів, налаштованих на обчислення вейвлет-перетворення кривих струму;

3) екрануванням мікропроцесора, вимірювального обладнання, проводів;

4) включення датчика струму – резистора Rд у коло загального проводу мікропроцесора. Цей підхід дозволяє значно знизити шум, оскільки осцилограф підключається до загального вивода всієї схеми (рис. 3.2);

5) усереднення результатів вимірювання на декількох повторюваних інтервалах.

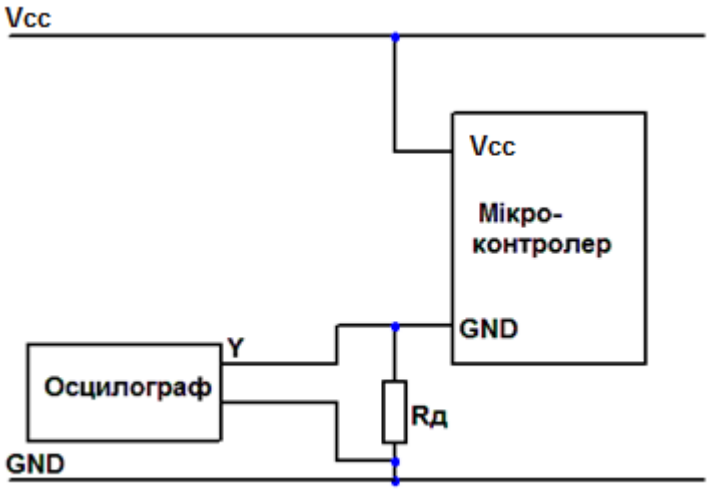


Рис.3.2. Схема підключення осцилографа для вимірювання струму споживання, що дозволяє мінімізувати рівень шумів.

Ще одним важливим параметром, що впливає на точність вимірювань є вхідна ємність щупа та самого осцилографа. Пасивні щупи мають більшу ємність, ніж активні. А чим більше вхідна ємність щупа, тим вужче смуга пропускання вимірюваного сигналу. Для зменшення вхідної ємності також можна використовувати додатковий посилювач на швидкодіючому малошумлячому операційному посилювачі. Зменшення довжини вимірювального кабелю, що підключається до осцилографа веде до зниження його вхідної ємності, але оскільки сучасні осцилографи використовують автоматичну калібровку і налаштовані на певну довжину вимірювального

кабелю, цей підхід може погіршити результати вимірювання. Тому важливим є вибір оптимальної довжини вимірювального кабелю

Використання постійного резистора у якості датчика струму має певні недоліки. Зокрема, перехідні струми можуть спричиняти значне падіння напруги на вимірювальному резисторі, і як наслідок, спричинити нестійку роботу досліджуваного мікропроцесора. Зменшення опору резистора вирішує дану проблему, але унеможливлює детальне дослідження малих за значенням перехідних струмів. Цих недоліків частково можна позбутися, використовуючи у якості датчика струму трансформатор. В цьому випадку можна легко отримати необхідну вихідну амплітуду сигналу просто зменшуючи чи збільшуючи кількість витків вторинної обмотки. Також трансформатор струму дозволяє позбавитися постійної складової, що може бути корисним при певних видах аналізу. Але використання трансформатора може спричинити звуження смуги пропускання сигналу, тобто в даному випадку трансформатор буде виконувати роль фільтра. Корисно мати можливість вимірювання струму двома вказаними датчиками, що дозволяє отримувати більше вхідних даних для аналізу.

3.2. Методика проведення експерименту та збору даних

При проведенні експерименту по оцінці ступеню захищеності мікроконтролера [52] [53] використовувалась експериментальна установка, структурна схема якої містить цифровий осцилограф, програматор та персональний комп’ютер (рис.3.3).

Рис.3.3. Структурна схема експериментальної установки.

Окремо слід відзначити обрання режиму вимірювання осцилографа «Oversampling Mode». В цьому режимі осцилограф вимірює періодичні сигнали з підвищеною роздільною здатністю за рахунок циклічного зміщення в часі моменту початку вимірювання АЦП відносно вхідного сигналу запуску. На рис.3.4. проілюстровано результати, що отримано даним способом вимірювання.

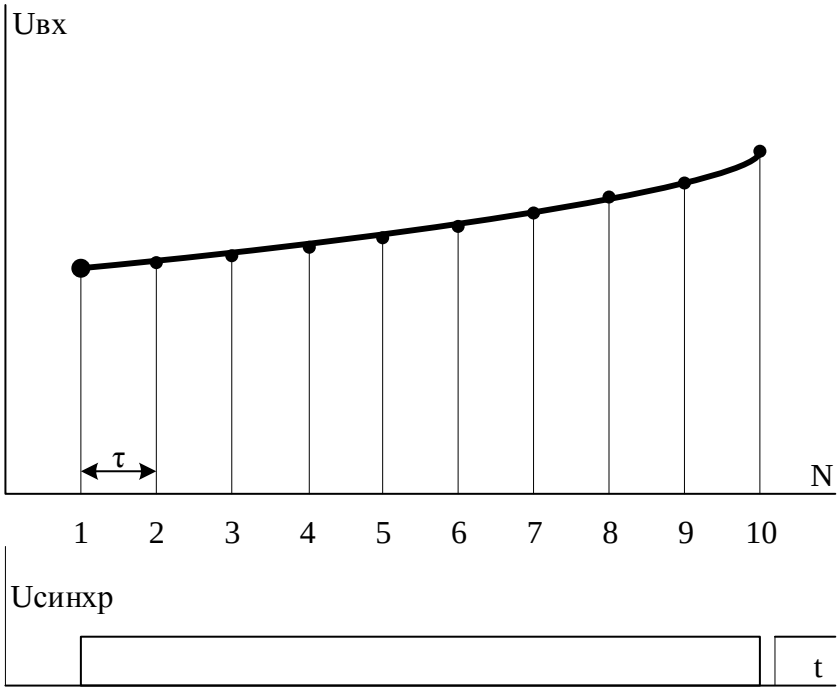


Рис.3.4 Вимірювання періодичного сигналу в режимі «Oversampling Mode». Uвх – вхідна напруга АЦП, пропорційна струму споживання; Uсинхр – напруга синхронізації; τ - часова затримка; цифрами показані номера вимірювань

Так, вимірювання №1 починається в момент часу після приходу переднього фронту імпульсу синхронізації $U_{синхр}$. Вимірювання №2 відбувається з часовою затримкою τ відносно надходження синхроімпульсу, вимірювання №3 – з затримкою 2τ і т.д. Результати кожного вимірювання зберігаються у пам’яті цифрового осцилографа, а потім відсортовуються у правильному порядку перед відображенням. Даний спосіб вимірювання дозволяє підвищити частоту дискретизації, однак може використовуватися лише для періодичних сигналів. Також необхідною умовою використання даного способу є наявність сигналу синхронізації. Однак такий спосіб вимірювання дозволяє отримати більше інформації про сигнал струму споживання, зокрема підвищити частоту дискретизації.

Після завантаження набору програм у пам'ять мікроконтролера виконувалося 10-ти кратне фіксація осцилограм струму для кожної

підпрограми, причому осцилограми зберігалися як у графічному вигляді, так і в текстовому. Приклад осцилограми струму споживання для інструкції CPSE наведено на рис.3.5. При цьому масштаб вимірювання встановлювався по кожному каналу окремо з таким розрахунком, щоб масштаб осцилограми був якомога більшим, але при цьому не було спотворень сигналу унаслідок його обмеження.

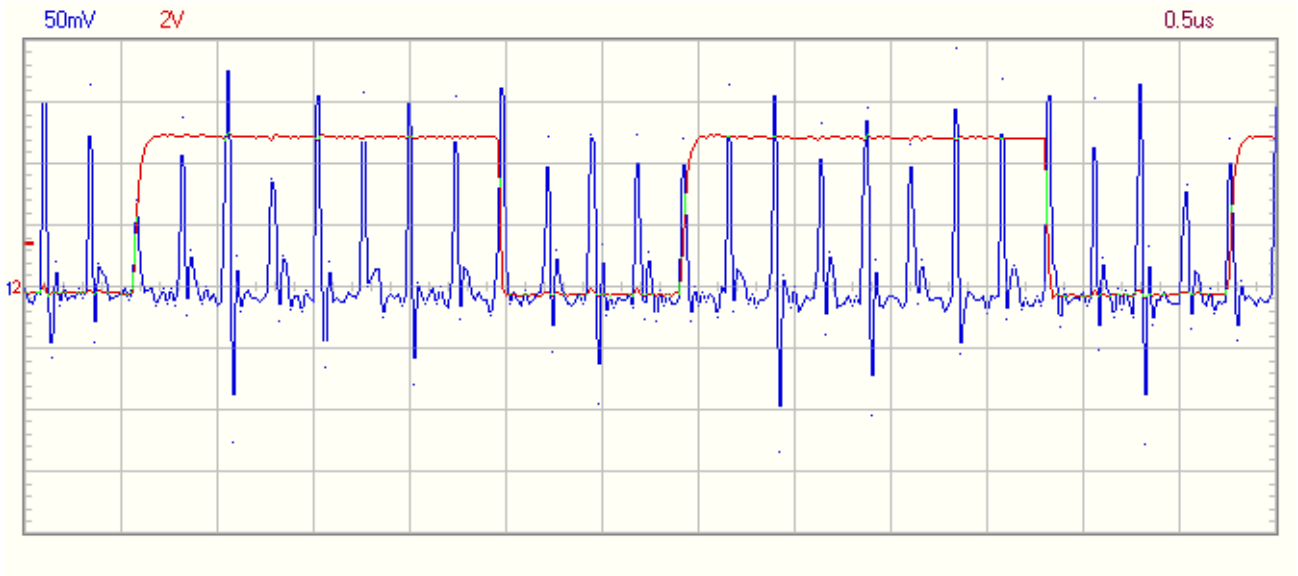


Рис.3.5 Приклад осцилограми струму в графічному вигляді для інструкції CPSE.

Результати усіх вимірювань зберігалися у файлах даних програми Oscilloscope. Такий файл має форма звичайного текстового файла, та може бути легко експортований в будь-яку програму обробки даних таку як Matlab, Mathcad, MS Excel або іншу. Розглянемо формат такого файлу на прикладі осцилограми струму в текстовому вигляді, що частково наведена в лістингу 3.1. Файл складається із заголовка (строчки 001-010) та блоку даних (011-021). В заголовку файла вказано часовий крок в строчці 003, при цьому вказується часовий крок для певної кількості точок. В даному випадку вказано, що 5 мкс відповідає 250 точкам. Тобто інтервал між двома сусідніми точками складає 20 нс, що відповідає частоті дискретизації 50МГц.

В строчках 006 та 007 вказано масштаб осцилограми по вісі напруг. В даному випадку рівень квантування між сусідніми точками складає 1.5625В та 0.0625В відповідно по 1-му каналу та по 2-му каналу. Строчка 008 містить значення рівня загального виводу. Цей рівень необхідно відняти від усіх наступних значень для того, щоб отримати значення сигналу відносно загального виводу. В строчках 011-016 міститься дискретні значення сигналу у відліках АЦП. Загальна кількість значень в одному файлі – 4096, що робить даний формат осцилограми набагато інформативнішим, ніж графічний формат представлення. Окрім того, текстовий формат даних дозволяє використовувати отримані дані з АЦП у самостійно розроблених програмах.

Лістинг 3.1

```
001
002     TIME STEP:
003     250 = 5us
004
005     VOLTAGE STEP:
006     CH1:  32 = 50mV
007     CH2:  32 = 2V
008     GND 126   127
009     N     CH1  CH2
010
011     0     103  204
012     1     126  124
013     2     119  123
014     3     118  124
015     4     124  124
016     5     123  124
```

Після отримання осцилограм у цифровому та графічному вигляді, вони зберігаються на диску для подальшої обробки. Після отримання та занесення у окремі файли осцилограм струмів для усіх команд мікропроцесора, створюється база даних команд та відповідних їм струмів споживання. Отримана база даних використовується як джерело вихідних даних для подальшої обробки та ідентифікації команд.

3.3. Дослідження струмів споживання за допомогою кореляційного аналізу

У якості критерію, що показує ступінь схожості струмів споживання між собою у випадку різних команд використовується коефіцієнт взаємної кореляції [29]. При цьому порівнюються коефіцієнти між однією парою або множиною пар ознак для встановлення між ними статистичної взаємодії.

Мета кореляційного аналізу – забезпечити отримання деякої інформації про одну змінну за допомогою іншої змінної [54] [55]. В самому загальному вигляді сприйняття гіпотези про наявність кореляції означає, що зміна значення змінної X відбудеться одночасно зі зміною значення Y.

Головні завдання кореляційного аналізу:

1) оцінка за вибірковими даними коефіцієнтів кореляції;
2) перевірка значущості вибіркових коефіцієнтів кореляції або кореляційного відношення;
3) оцінка близькості виявленого зв’язку до лінійного;
4) побудова довірчого інтервалу для коефіцієнтів кореляції.

Кореляція відображає лише лінійну залежність величин, але не відображає їх функціональної зв’язаності. Наприклад, якщо обчислити коефіцієнт кореляції між величинами X = sin(x) та Y = cos(x) з випадковим фазовим зсувом, він може бути наближений до нуля, тобто залежність між величинами відсутня. Між тим, величини X та Y очевидно зв’язані між собою за законом $sin^2(x) + cos^2(x) = 1$.

Кореляційний зв'язок представляє собою частковий випадок статистичного зв'язку $M(Y|X=x) = y(x)$, тобто математичне очікування змінної Y, при умові, що випадкова величина X приймає значення x.

Стохастичним зв'язком між випадковими величинами називається такий зв'язок, при якій із зміною однієї величини змінюється розподіл іншої. Функціональною залежністю називається така зв'язок між випадковими величинами, при якій при відомому значенні однієї з величин можна точно вказати значення іншої. На відміну від функціональної зв'язку при

стохастичному зв'язку зі зміною величини X величина Y має лише тенденцію змінюватися. У міру збільшення тісноти стохастичної залежності вона все більш наближається до функціонального. Крайня протилежність функціонального зв'язку – повна незалежність випадкових величин. Якщо випадкові величини незалежні, то згідно з теоремою множення одержуємо:

$$f(y/x) = f2(y) \text{ і } f(x/y) = f1(x), \quad (3.1)$$

$$f(x, y) = f1(x) \cdot f2(y). \quad (3.2)$$

Умову (3.2) можна використовувати в якості необхідного і достатньо точного критерію незалежності двох випадкових величин, якщо відомі щільності розподілу системи і випадкових величин, що входять до неї.

При невідомому законі розподілу системи для оцінки тісноти стохастичною зв'язку найчастіше використовується коефіцієнт кореляції. Дисперсія суми двох випадкових величин X і Y дорівнює:

$$D(X+Y) = M([X+Y-M(X+Y)]^2) = M([X-M(X)+Y-M(Y)]^2) =$$
$$= M[X-M(X)]^2 + 2M([X-M(X)]\cdot[Y-M(Y)]) + M[Y-M(Y)]^2 =$$
$$= D(X) + 2M([X-M(X)]\cdot[Y-M(Y)]) + D(Y). \quad (3.3)$$

Якщо X і Y незалежні, то:

$$D(X+Y) = D(X) + D(Y). \quad (3.4)$$

Тоді залежність між X та Y існує, якщо:

$$M([X-m_x]\cdot[Y-m_y]) \neq 0. \quad (3.5)$$

Величина (3.5) є кореляційним моментом, або коваріацією cov(XY), випадкових величин. Вона характеризує не тільки залежність величин, але і їх розсіювання. З (3.5) випливає, що якщо одна з величин мало відхиляється від свого математичного сподівання, то коваріація буде мала навіть при тісному стохастичною зв'язку. Щоб уникнути цього, для характеристики зв'язку слід використовувати безрозмірну величину – коефіцієнт кореляції:

$$r_{xy} = \frac{\mathrm{cov}_{xy}}{\sigma_x \sigma_y} = \frac{M\left(\left[X-m_x\right]\left[Y-m_y\right]\right)}{\sigma_x \sigma_y}, \quad (3.6)$$

де σX, σY — стандартне відхилення величин X та Y.

Якщо X та Y — незалежні, то коефіцієнт кореляції дорівнює 0. Зворотне твердження невірне. Коефіцієнт кореляції може дорівнювати 0, навіть якщо Y є функцією від X.

Завжди виконується нерівність: $|\rho(X,Y)| \le 1$.

Причому, $\rho(X,Y) = \pm 1$ тоді і лише тоді, коли $y = ax + b$, де a та b – сталі.

Відмітимо наступні властивості коефіцієнту кореляції:

1) величина r_{xy} не міняється від додавання до X і Y невипадкових доданків;

2) величина r_{xy} не змінюється від множення X і Y на позитивниі числа;

3) якщо одну з величин, не змінюючи інший, помножити на –1, то на –1 помножиться і коефіцієнт кореляції.

Якщо коефіцієнт кореляції обчислюється між змінними, що належать порядковій шкалі, застосовується коефіцієнт Спірмена, а для змінних, що належать до інтервальної шкали - коефіцієнт кореляції Пірсона (момент добутків). Інтерпретація отриманих результатів відбувається за табл.3.1.

Таблиця. 3.1

Значення коефіцієнта кореляції	Інтерпретація
0 < r <= 0,2	дуже слабка кореляція
0,2 < r <= 0,5	слабка кореляція
0,5 < r <= 0,7	середня кореляція
0,7 < r <= 0,9	сильна кореляція
0,9 < r <= 1	дуже сильна кореляція

Коефіцієнт кореляції Пірсона використовується якщо:

- Усі спостереження взаємно незалежні.
- Спостереження мають нормальний закон розподілу.

Значення коефіцієнта кореляції обчислюється за формулою:

$$p_{xy} = \frac{\sum_{i=1}^{N}\left((X_i - \overline{X})\cdot(Y_i - \overline{Y})\right)}{\sqrt{\sum_{i=1}^{N}(X_i - \overline{X})^2 \cdot \sum_{n=0}^{N-1}(Y_i - \overline{Y})^2}} \qquad (3.7)$$

Оскільки проведений експеримент відповідає зазначеним критеріям [54], то розрахунок коефіцієнтів кореляції здійснюється за формулою (3.7).

Порівнюючи за допомогою коефіцієнта взаємної кореляції струмів споживання між собою, можна зробити висновок про ступінь схожості еталонного значення струму та значення струму, що вимірюється. Це дозволяє детектувати мікропроцесорні команди, значення струму яких вибирається із заздалегідь створеної бази даних.

Оцінка кореляції відбувається у часовій області, у області вейвлетів, та у частотній області за допомогою швидкого перетворення Фур'є. Необхідність дослідження кореляції у вказаних областях викликане необхідністю пошуку оптимального та ефективного алгоритму визначення мікропроцесорної команди по її струму споживання.

3.4. Розробка програмного забезпечення у системах C++ та MatLab

Для дослідження та обробки отриманих у ході експериментів даних створено дві програми. Перша програма «Correlation» призначена для зчитування текстових файлів з даними та наочного представлення цих даних у вигляді таблиць. Це дозволяє швидко визначити необхідні початкові точки послідовності та довжину послідовності для порівняння. Крім того, в програмі можливо оцінити кореляцію двох послідовностей з двох різних файлів або з одного файлу. Програма написана з використанням Visual C++ [56] [57] [58] [59]. Зовнішній вигляд головного вікна програми наведений на рис 3.6. Текст програми, що виконує обчислення коефіцієнту взаємної кореляції за формулою (3.7) наведено у лістингу 3.2.

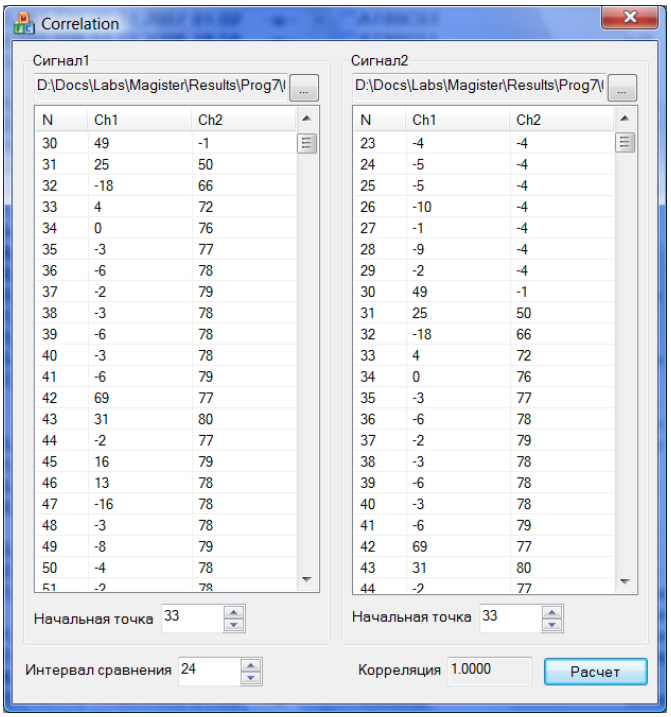


Рис 3.6. Зовнішній вигляд програми “Correlation”

Лістинг 3.2

```
001   Void CCorrelationDlg::DoCorrelation(void)
002   {

003         if (m_pDataA1 == NULL || m_pDataB1 == NULL)
004               return;

005         int nStart1=0;
006         int nStart2=0;
007         int nRange=0;

008         CString tmp;
009         m_edStart1.GetWindowText(tmp);
010         tmp.Replace("\x0A0","");
011         nStart1=atoi(tmp);

012         m_edStart2.GetWindowText(tmp);
013         tmp.Replace("\x0A0","");
014         nStart2=atoi(tmp);

015         m_edRange.GetWindowText(tmp);
016         tmp.Replace("\x0A0","");
017         nRange=atoi(tmp);

018         int k,j,i,r0,ra,rb;
019         double norm, r0final;

020         BOOL bResultOk=true;

021         r0=0;
022         ra=0;
```

```
023         rb=0;

024         i=nStart1;
025         j=nStart2;

026         for(k=0;k<nRange;k++)
027         {
028               if (i>4095 || j>4095)
029               {
030                     bResultOk = false;
031                     break;
032               }

033               r0+=m_pDataA1[i]*m_pDataB1[j]; /// needs to be changed here;
034               ra+=m_pDataA1[i]*m_pDataA1[i];
035               rb+=m_pDataB1[j]*m_pDataB1[j];
036               i++;
037               j++;
038         }

039         if (bResultOk)
040         {

041               norm=pow((double)ra*(double)rb,(double)0.5);


042               r0final=(double)r0/(double)norm;


043               tmp.Format("%.4lf",r0final);
044         }
045         else
046         {
047               tmp = "ERROR";
048         }

049         m_edCorr.SetWindowText(tmp);

050   }
```

Але, оскільки оцінка та порівняння між собою великого масиву даних в ручному режимі вимагає багато часу, для спрощення цієї задачі написано програму в системі Matlab [60] [61] [62]. Використання системи Matlab дозволяє значно спростити вихідний текст програми за наявності великої кількості вбудованих функцій для роботи з векторами, матрицями, та математичними виразами. Перевагою системи MatLab [63] є досить проста візуалізація даних у вигляді двовимірних та тривимірних графіків, поверхонь. Обробка великих масивів даних може займати багато часу навіть на сучасних

комп'ютерах. Саме цей недолік і перешкоджає використанню MatLab у системах обробки даних реального часу.

Програма для визначення коефіцієнта взаємної кореляції між струмами мікропроцесора написана для системи MatLab та складається з основної програми на набору файл-функцій, які виконують допоміжні обчислення та обробку даних для основної програми.

Файл-функція get_start.m, наведена в лістингу 3.3, використовується для виділення з файлового масиву даних послідовності за тактовими імпульсами.

Лістинг 3.3.

```
001    function f = get_start(filename, np)

002    F=fopen(filename,'rt');

003    line = '';

004    for i = 1:7
005        line = fgetl(F);
006    end

007    line = fscanf(F,'%s',1);

008    g1 = fscanf(F,'%d',1);
009    g2 = fscanf(F,'%d',1);

010    line = fgetl(F);
011    line = fgetl(F);

012    N(1:4096)=0;
013    ML1(1:4096)=0;
014    ML2(1:4096)=0;

015    for i=1:4096
016    N(i) = fscanf(F,'%d',1);
017    l1 = fscanf(F,'%d',1);
018    l2 = fscanf(F,'%d',1);
019    ML1(i)=l1-g1;
020    ML2(i)=l2-g2;
021    end
022    fclose(F);


023    i=10;
024    npx=1;

025    f=-1;

026    while(1)
027        while(ML2(i)<20)
028            if (i>=4096)
```

```
029                 break
030             end
031             i=i+1;
032         end

033         i=i+2;

034         if (npx==np)
035             f=i;
036             break
037         end
038         npx=npx+1;
039         while(ML2(i)>=0)
040             if (i>=4096)
041                 break
042             end
043             i=i+1;
044         end
045     end
```

В якості параметрів (рядок 001) до файл-функції передається ім’я файлу та номер послідовності в файлі, яку треба відшукати. Файл-функція відкриває вказаний файл та переходить до 8-ї строчки цього файла (рядки 002-006). В рядках 007-011 файл-функція зчитує значення нульових рівнів для обох каналів. Ці значення зберігаються для подальшої корекції всіх зчитуваних даних на рівень нульового виводу. В рядках 012-022 відбувається порядкове зчитування вмісту відкритого текстового файлу з занесенням значень у рядки матриці. При занесенні кожного значення відбувається його корекція на величину нульового рівня, який було зчитано раніше. В рядках 026-045 організовано цикл пошуку позитивного перепаду сигналу на другому каналі. Якщо такий перепад знайдено, значення функції набуває номеру точки у вихідній послідовності. У випадку, коли позитивний перепад не знайдено на 2-му каналі, результатом виконання функції буде від’ємне число.

В лістингу 3.4 показана файл-функція для системи Matlab, що використовується для виділення з загальної файлової бази певного фрагменту, за вказаним ім’ям файла, положенням у цьому файлі та довжиною фрагменту.

Лістинг 3.4.

```
001   function f = get_cmd(filename, beg, len)

002   F=fopen(filename,'rt');

003   line = '';

004   for i = 1:7
          line = fgetl(F);
005   end

006   line = fscanf(F,'%s',1);

007   g1 = fscanf(F,'%d',1);
008   g2 = fscanf(F,'%d',1);

009   line = fgetl(F);
010   line = fgetl(F);

011   for i=1:4096
012   N(i) = fscanf(F,'%d',1);
013   l1 = fscanf(F,'%d',1);
014   l2 = fscanf(F,'%d',1);
015   ML1(i)=l1-g1;
016   ML2(i)=l2-g2;
017   end
018   fclose(F);

019   f=ML1((beg+1):(beg+len));
```

Лістинг цієї файл- функції відрізняється від попередньої тим, що в рядку 019 відбувається виділення знайденого фрагменту та передача цього масиву до головної програми.

Лістинг 3.5 містить код головної програми, що використовує вищенаведені файл-функції та виконує значну частину обробки даних. Рядки 001-005 містять список текстових файлів, що потребують порівняння.

Лістинг 3.5

```
001   findpath = 'D:\docs\labs\magister\results\';

002   FILES1 = char('prog1\4-1.txt', 'prog1\6-1.txt','prog1\9-1.txt','prog1\10-
      1.txt','prog1\11-1.txt','prog1\13-1.txt',...
003       'prog1\14-1.txt','prog1\2-1.txt','prog2\0-1.txt','prog2\1-
      1.txt','prog2\7-1.txt','prog2\8-1.txt','prog2\9-1.txt',...
004       'prog2\10-1.txt','prog2\11-1.txt','prog2\12-1.txt','prog2\13-
      1.txt','prog2\14-1.txt','prog2\15-1.txt',...
005       'prog3\0-1.txt','prog3\1-1.txt','prog3\4-1.txt','prog2\3-
      1.txt','prog2\4-1.txt','prog2\5-1.txt','prog2\6-1.txt');
```

```
006   n = 26;
007   len = 24;

008   STARTS1(1:n)=0;
      STARTS2(1:n)=0;

009   for i=1:n
010       STARTS1(i)=get_start(deblank([findpath FILES1(i,:)]),1);
011       STARTS2(i)=get_start(deblank([findpath FILES1(i,:)]),2);
012
013   end
014   CORRTIME(1:n,1:n)=0;
015   CORRFOURIER(1:n,1:n)=0;
016   CORRWHAAR(1:n,1:n)=0;
017   CORRWOB(1:n,1:n)=0;

018   start1=0;
019   start2=0;

021   for i=1:n
022       for j=i:n
023
024       if (i==j)
025           start1=STARTS1(i);
026           start2=STARTS2(i);
027       else
028           start1=STARTS1(i);
029           start2=STARTS1(j);
030       end

031       D1 = get_cmd(deblank([findpath FILES1(i,:)]),start1,len);
032       D2 = get_cmd(deblank([findpath FILES1(j,:)]),start2,len);
033       CORRTIME(i,j)=corr2(D1,D2);

034       DF1=fft(D1);
035       DF2=fft(D2);
          CORRFOURIER(i,j)=corr2(abs(DF1),abs(DF2));

036       [cA1,cD1] = dwt(D1,'haar');
037       [cA2,cD2] = dwt(D2,'haar');

038       CORRWHAAR(i,j)=corr2(cD1,cD2);

039       [cA1,cD1,cL1] = waveob(D1);
040       [cA2,cD2,cL2] = waveob(D2);



041       CORRWOB(i,j)=corr2([cD1 cL1],[cD2 cL2]);

042       end
043   end
```

Після завершення обчислень наведена програма формує таблиці кореляційних коефіцієнтів для кожного методу:

1) CORRTIME – таблиця коефіцієнтів взаємної кореляції при порівнянні у часовій області

2) CORRFOURIER - таблиця коефіцієнтів взаємної кореляції при порівнянні у частотній області за допомогою швидкого перетворення Фур'є.

3) CORRWHARR - таблиця коефіцієнтів взаємної кореляції при порівнянні коефіцієнтів деталізації після виконання вейвлет-перетвореня Хаара.

4) CORRWOB - таблиця коефіцієнтів взаємної кореляції при порівнянні коефіцієнтів деталізації після виконання ОБ вейвлет-перетвореня.

Оскільки таблиці симетричні відносно головної діагоналі, для скорочення часу обчислень, обчислюються лише коефіцієнти вище головної діагоналі та ті, що знаходяться в головній діагоналі. Інші коефіцієнти не обчислюються.

В лістингу 3.6 наведена програма для порівняння даних, що міститься у вищенаведених таблицях та виявлення методів, за допомогою яких можливо виділити окремі команди.

Лістинг 3.6

```
001   TOTS = cell(n);
002   bOpr=0;
003   NT=0;
004   NF=0;
005   NH=0;
006   NO=0;
007   NX=0;
008   NTOTAL=0;

009   for i=1:n
010       for j=i:n

011           TOTS{i,j}='';
012           bOpr=0;
013           NTOTAL = NTOTAL+1;

014           if (CORRTIME(i,j)<0.9)
015               TOTS{i,j}=[TOTS{i,j} 'T'];
016               bOpr=1;
017               NT = NT+1;
018           end

019           if (CORRFOURIER(i,j)<0.9)
```

```
020                 TOTS{i,j}=[TOTS{i,j} 'F'];
021                 bOpr=1;
022                 NF = NF+1;
023             end

024             if (CORRWHAAR(i,j)<0.9)
025                 TOTS{i,j}=[TOTS{i,j} 'H'];
026                 bOpr=1;
027                 NH = NH+1;
028             end

029             if (CORRWOB(i,j)<0.9)
030                TOTS{i,j}=[TOTS{i,j} 'O'];
031                bOpr=1;
032                NO = NO+1;
033             end

034             if (bOpr == 0)
035                 if (i ~= j)
036                     NX = NX+1;
037                 end
038                 TOTS{i,j}='x';
039             end
040         end
041     end
042     PT = 100 * NT / NTOTAL;
043     PF = 100 * NF / NTOTAL;
044     PH = 100 * NH / NTOTAL;
045     PO = 100 * NO / NTOTAL;
046     PX = 100 * NX / NTOTAL;
```

Наведена програма оцінює коефіцієнти взаємної кореляції у всіх комірках таблиць CORRTIME (рядки лістингу 014-018), CORRFOURIER (рядки лістингу 019-023), CORRWHAAR (рядки лістингу 024-028), CORRWOB (рядки лістингу 029-033), в тому випадку, якщо коефіцієнт кореляції у комірці менший 0.9, програма робить відповідну помітку в таблиці TOTS (табл. 3.2). Ці помітки можуть приймати одне або кілька з нижченаведених значень:

1) «T» - дані дві команди можуть бути розрізнені після порівняння їх струмів споживання у часовій області.

2) «F» - дані дві команди можуть бути розрізнені після порівняння їх струмів споживання у частотній області.

3) «H» - дані дві команди можуть бути розрізнені після вейвлет-розкладання їх струмів споживання за допомогою вейвлетів Хаара та порівняння відповідних коефіцієнтів деталізації.

4) «О» - дані дві команди можуть бути розрізнені після вейвлет-розкладання їх струмів споживання за допомогою ОБ-вейвлет перетворенн та порівняння відповідних коефіцієнтів деталізації.

5) «Х» - дані дві команди не можливо розрізнити жодним з використаних способів порівняння.

В рядках лістингу 017, 022, 027, 032 підраховується кількість успішних порівнянь двох струмів споживання, в результаті яких ці струми виявляються різними. В ідеальному випадку необхідно отримати схожі струми лише на головній діагоналі табл. 3.2, а усі інші елементи цієї таблиці повинні прямувати до 0.

Таблиця 3.2 Детектування команд на етапах

	CPI	BREQ	BRNE	CP(Z=1)	CP(Z=0)	BRCC	BRCS	CPSE	MOV	LDI	CLZ	ROL	ROR	SWAP	ADD	SUB	CLR	SER	COM	NEG	BRMI	BRPL	NOP	SEC	CLC	SEZ
CPI	x	TF	TFH	x	TFHO	TFHO	x	THO	TFHO	TF	TFH	TFH	TFH	x	TFHO	TFHO	x	TFHO	x	TFHO	THO	THO	THO	x	x	x
BREQ		x	x	TF	THO	THO	x	THO	THO	x	x	x	x	TF	THO	THO	x	THO	x	THO	TFHO	TFHO	THO	x	x	x
BRNE			x	TFH	THO	THO	x	TFHO	THO	x	x	x	x	TFH	THO	THO	F	THO	x	THO	TFHO	TFHO	TFHO	x	x	x
CP(Z=1)				x	TFHO	TFHO	F	THO	TFHO	TF	TFH	TF	TFH	x	TFHO	TFHO	x	TFHO	x	TFHO	HO	THO	THO	x	x	x
CP(Z=0)					x	x	THO	TFH	x	THO	THO	THO	THO	TFHO	THO	x	TFHO	x	THO	x	TFHO	TFHO	TFH	THO	THO	THO
BRCC						x	THO	TFH	x	THO	THO	THO	THO	TFHO	THO	x	TFHO	x	THO	x	TFHO	TFHO	TFH	TFHO	TFHO	TFHO
BRCS							x	THO	THO	x	x	x	x	x	THO	THO	x	THO	x	THO	THO	TFHO	THO	x	x	x
CPSE								x	TFH	THO	TFHO	TFHO	THO	THO	THO	TFHO	THO	H	THO	H	HO	H	x	THO	THO	THO
MOV									x	THO	THO	THO	THO	TFHO	THO	x	TFHO	x	THO	x	TFHO	TFHO	TFH	TFHO	THO	TFHO
LDI										x	x	x	x	TFH	THO	THO	x	THO	x	THO	TFHO	TFHO	THO	x	x	x
CLZ											x	x	x	TFH	THO	THO	F	THO	x	THO	TFHO	TFHO	TFHO	x	x	x
ROL												x	x	TFH	THO	THO	F	THO	x	THO	TFHO	TFHO	TFHO	x	x	x
ROR													x	TFH	THO	THO	x	THO	x	THO	TFHO	TFHO	THO	x	x	x
SWAP														x	TFHO	TFHO	x	TFHO	x	TFHO	THO	THO	THO	x	x	x
ADD															x	THO	THO	THO	THO	THO	TFHO	TFHO	THO	THO	THO	THO
SUB																x	TFHO	x	THO	x	TFHO	TFHO	TFHO	TFHO	TFHO	TFHO
CLR																	x	THO	x	TFHO	THO	THO	THO	x	x	x
SER																		x	THO	x	TFHO	TFHO	H	THO	THO	THO
COM																			x	THO	THO	THO	THO	x	x	x
NEG																				x	TFHO	TFH	H	THO	THO	THO
BRMI																					x	x	x	THO	THO	THO
BRPL																						x	x	THO	THO	THO
NOP																							x	THO	THO	THO
SEC																								x	x	x
CLC																									x	x
SEZ																										x

3.5. Обробка результатів обчислень

Обробка результатів обчислень дозволяє оцінити ефективність детектування команд для різних методів обробки, причому команда вважається ідентифікованою при заданому значенні коефіцієнта взаємної кореляції.

У результаті виконання порівняльного аналізу струмів споживання за допомогою коефіцієнта взаємної кореляції одержано показники ефективності, як частки визначених команд, для використаних методів детектування (рис 3.7):

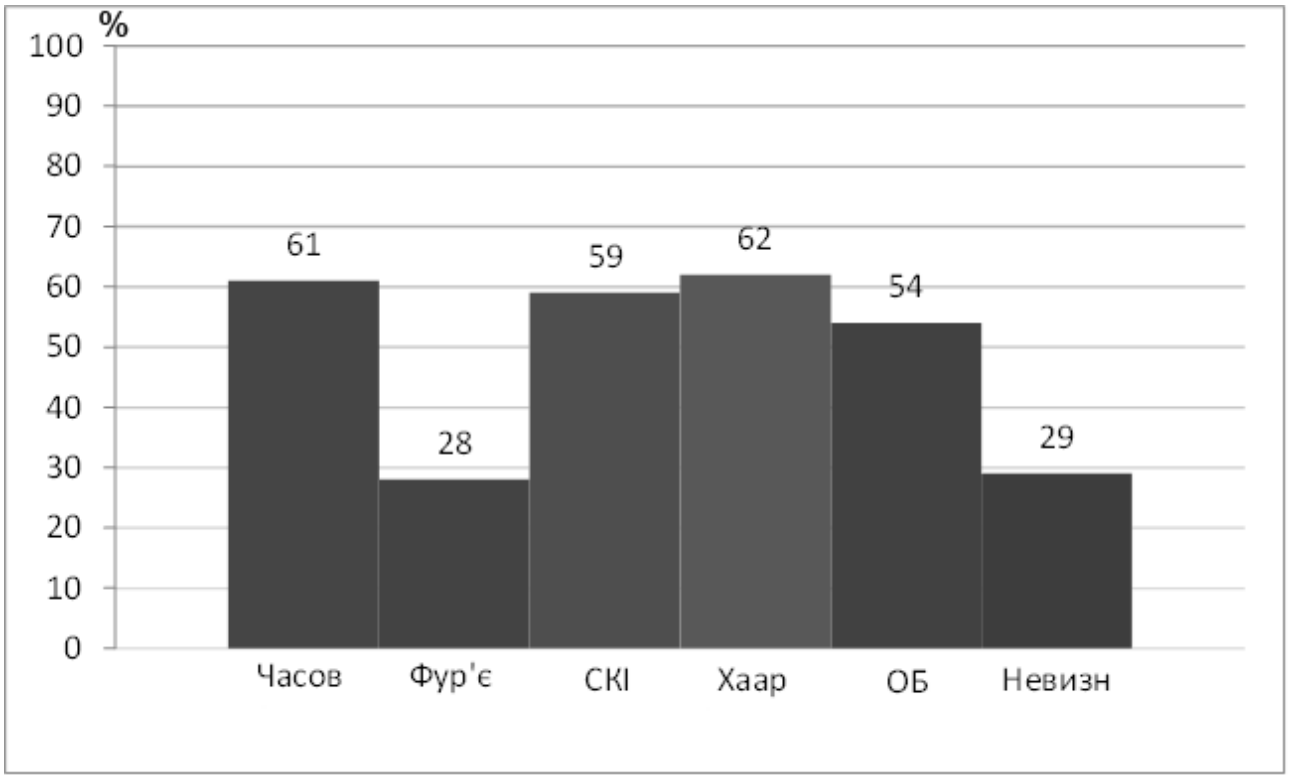


Рис.3.7. Ефективність методів детектування, %

Ефективність детектування оцінюється для кожного методу як відношення розпізнаних команд до загальної кількості у відсотках. (рядки 042-046 лістингу 3.6)

Очевидно, що для ідентифікації більшості команд (61%) достатньо використання порівняння струмів споживання у часовій області. Дещо підвищити ефективність даного методу можна за умови використання вейвлет-перетворень Хаара та ОБ (відповідно 62% та 54%). Низькою ефективністю у порівнянні з іншими методами володіє Фур’є перетворення (28%), оскільки воно дає менше інформації про миттєві значення сигналу, і призначене для

дослідження періодичних сигналів. У 29% випадків порівняння струмів різних команд не виявило між ними відмінностей.

Незважаючи на те, що метод порівняння у часовій області є досить ефективним, у деяких випадках, вказаних в таблиці, команди відокремити можна лише після проведення порівняння у області вейвлетів, це стосується зокрема відокремлення команд BRMI, BPRPL, CPSE, COM, NEG та деяких інших.

Враховуючи вищесказане, можна зробити висновок про те, що визначення команди по довільному струму споживання слід виконувати в декілька етапів, на кожному етапі обчислюючи кореляцію з відповідними значеннями з бази даних струмів. Перший етап детектування має складатися з порівняння струму споживання у часовій області з відомими значеннями струму. В тому випадку, якщо варіантів команди буде декілька, слід продовжити відсіювання, обчислюючи кореляцію між коефіцієнтами деталізації після виконання вейвлет-перетворення. В тому випадку, якщо кількість можливих варіантів команд залишиться більше одного, слід виконати порівняння в частотній області, застосувавши до обох послідовностей (шуканої та еталонної) перетворення Фур'є, а потім оцінити коефіцієнти взаємної кореляції.

Слід зазначити, що деякі команди не можливо відрізнити від інших схожих команд, оскільки такі команди відрізняються лише прапорцями. Наприклад, команди BRPL, BRMI, BRCS, BRCC, CSPE відрізняються лише прапорцями,, що унеможливлює їх детектування. Це зауваження стосується також команд встановлення прапорців, таких як CLZ, SEZ, SEC, CLC. Ще однією проблемою, яка постає на шляху визначення асемблерного коду виконуваної програми є визначення аргументів команд. В даному дослідженні переважно використовувалися команди які не приймають аргументів, або команди, аргумент яких не змінюється в ході виконання програми.

Отже, за допомогою методу дослідження струму споживання мікроконтролера, у загальному випадку неможливо отримати вихідний

асемблерний код програми. Однак можна робити висновки про те, які алгоритми в даний момент виконує мікроконтролер, наприклад алгоритми обчислення, вводу-виводу даних, програмування та ін.

Таким чином, враховуючи особливості мікросхем, побудованих за технологією КМДН, вимірювати необхідно їх динамічний струм споживання, оскільки статичний струм споживання таких мікросхем практично дорівнює нулю.

Результатом детектування в загальному випадку є не одна конкретна команда процесора, а набір команд, які генерують схожі струми споживання.

РОЗДІЛ 4

ПОБУДОВА РЕГУЛЬОВАНИХ ФІЛЬТРІВ ІЗ МАСКУВАННЯМ СТРУМУ СПОЖИВАННЯ

При розробці регульованих фільтрів джерел живлення мікропроцесорних систем із маскуванням струму споживання необхідно вирішити дві задачі: 1) забезпечення якомога більшого рівня захисту, та 2) економічна доцільність реалізації фільтру на сучасній елементній базі.

4.1. Побудова регульованих фільтрів із змінними параметрами з використанням генератора шуму

В розділі 1 розглянуто основні шляхи реалізації систем керування фільтрами живлення із захистом інформації. На базі проведеного дослідження можливо побудувати нові фільтри живлення із змінними параметрами, які характеризуються підвищенням рівня захищеності та відносно простою реалізацією.

Місце регульованого фільтру в структурі системи перетворення електроенергії для живлення [64] мікропроцесорної системи зображене на рис.4.1. Мікропроцесорна система [65], що захищається, підключається до вторинного напівпровідникового джерела живлення через регульований фільтр з маскуванням струму споживання. Для запобігання несанкціонованого відокремлення регульованого фільтру від мікропроцесорної система, вони мають знаходитись всередині корпусу з детектуванням відкриття.

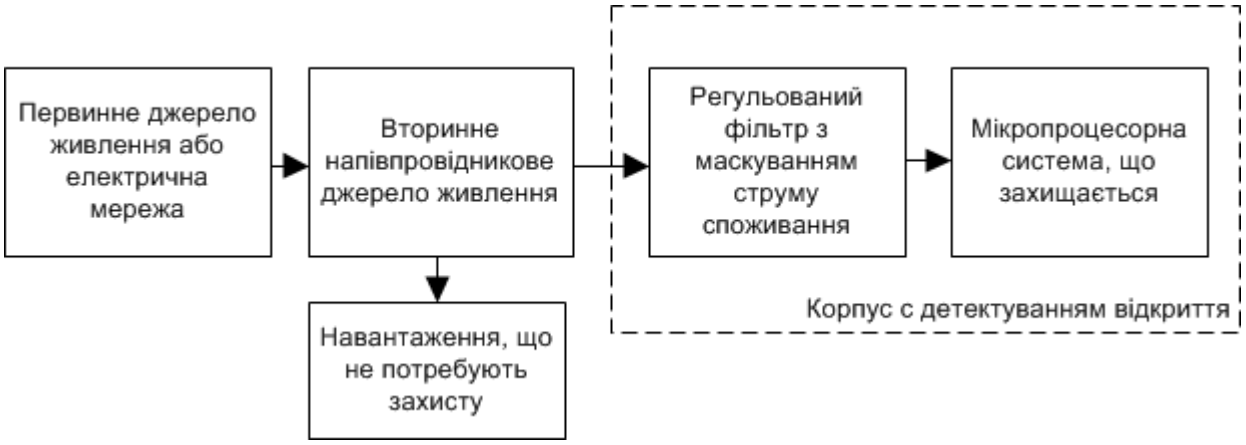


Рис.4.1. Структурна схема з перетворювачем електроенергії для живлення мікропроцесорної системи з маскуванням струму споживання

Еквівалентна схема регульованого фільтру із змінними параметрами, представлена на рис.4.2. Для RC-фільтру, опір R або ємність C змінюються за законом випадкових чисел, при цьому важливо визначити, який тип розподілення генератора випадкових чисел слід вибрати для забезпечення високого рівня захисту від зчитування інформації за струмом споживання.

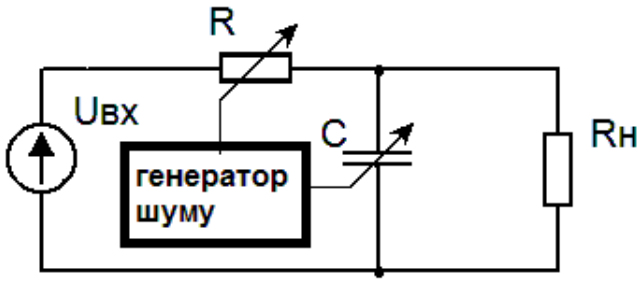


Рис.4.2. Еквівалентна схема регульованого фільтру із змінними параметрами

Для визначення типу розподілення генератора випадкових чисел запропонована спрощена модель системи живлення мікропроцесора з регульованим фільтром у системі Matlab-Simulink [66] [67] [68] [69] [70].

Модель (рис.4.3) складається з моделі джерела живлення (DC Voltage Source), послідовно включеного опору (Resisitor 1), який моделює опір джерела

живлення, опору навантаження (Resistor 2), яке забезпечує моделювання постійної складової струму споживання мікропроцесора, та змінного навантаження (Variable Resisitor 1), яке моделює змінну складову струму споживання мікроконтролера.

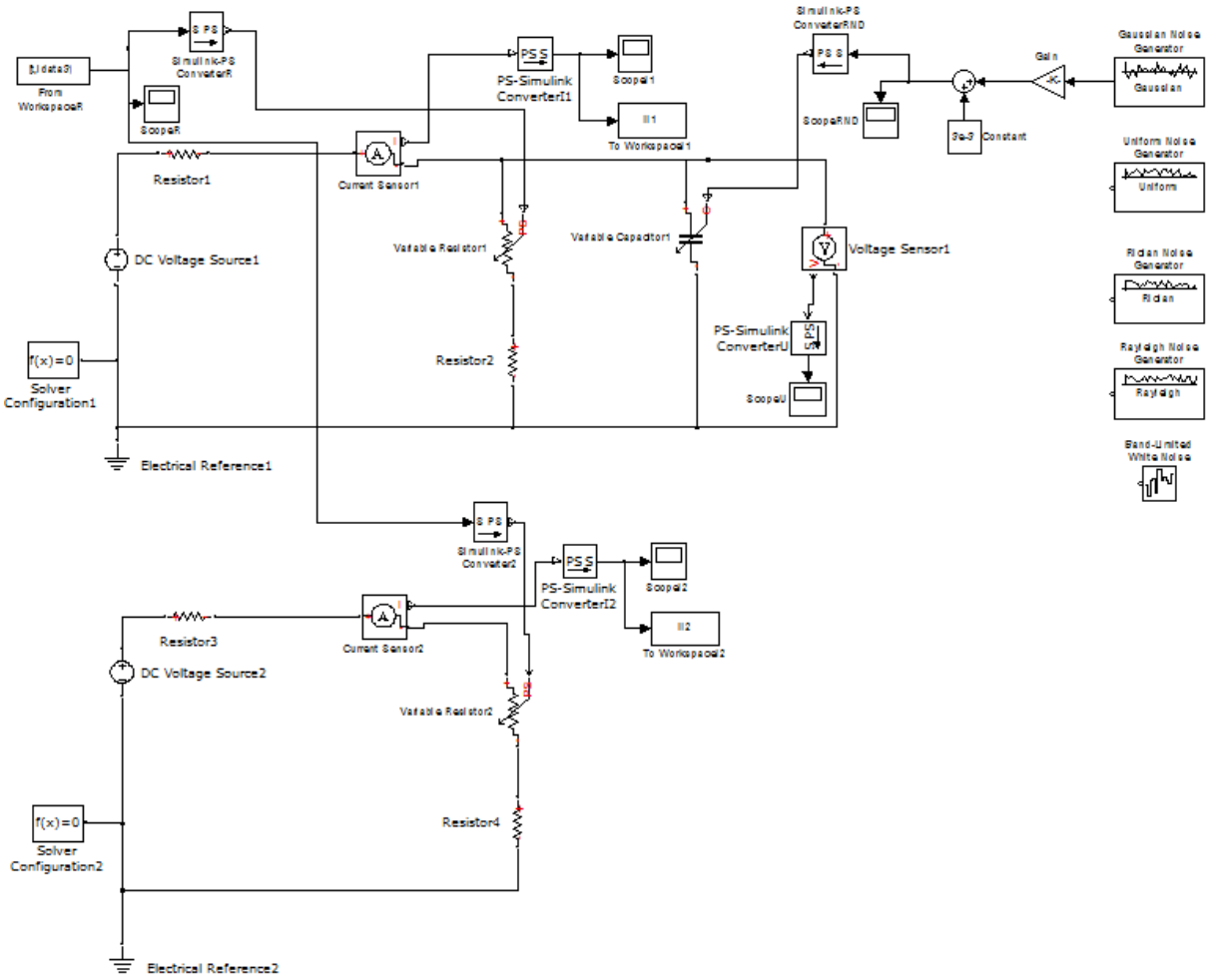


Рис.4.3. Модель регульованого фільтру зі змінним конденсатором

Керування змінним опором (Variable Resisitor 1) відбувається так, щоб змоделювати струм споживання реального мікроконтролера під час виконання команди.

Паралельно до навантаження підключено змінний конденсатор (Variable Capacitor 1). Значення ємності змінного конденсатора задається з виходу одного з генераторів шуму. Між генератором шуму та входом керування змінним конденсатором включено підсилювач та блок додавання постійної складової для того, щоб ємність конденсатора завжди була більше нуля.

Модель також містить схему навантаження зі змінним опором, але без включеного паралельно змінного конденсатора, для того, щоб можливо було виконати порівняння струму споживання у колі живлення без будь-якого захисту, та із включеним паралельно регульованим фільтром із змінним конденсатором.

Отримані струми споживання відображаються за допомогою осцилографів та записуються у змінні для подальшого аналізу.

Струм споживання моделі мікроконтролера без включення регульованого фільтру зображено на рис.4.4.

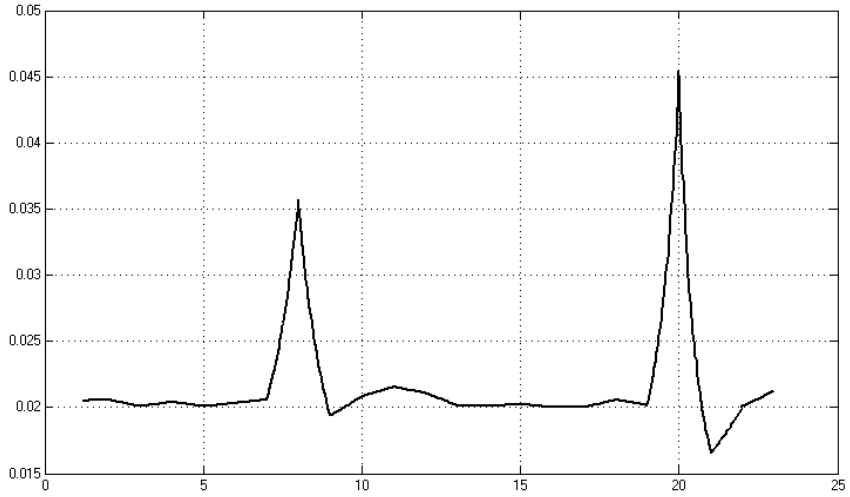


Рис.4.4. Струм споживання моделі мікроконтролера без регульованого фільтру

Струм споживання моделі мікроконтролера з паралельним включенням регульованого фільтру зображений на рис.4.5. У порівнянні зі струмом споживання без використання регульованого фільтру, в струмі з’явилися додаткові стрибки струму, що зменшує імовірність його детектування.

Було проведено моделювання з різними частотами та типами розподілень генераторів випадкових чисел. В процесі оброблення результатів моделювання було виявлено, що чим більша частота генератора випадкових чисел у порівнянні з частотою тактування мікроконтролера, тим більше спотворюється

струм споживання, і тим менша імовірність детектування команд. Так, на рис.4.6. зображено струм споживання моделі мікроконтролера з частотою генератора білого шуму, в 10 раз вищою, ніж частота тактування.

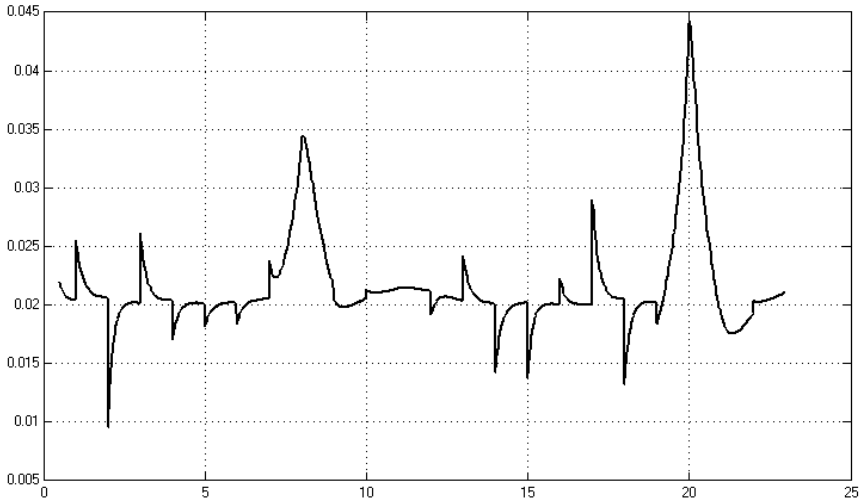


Рис.4.5. Струм споживання моделі мікроконтролера з регульованим фільтром з використанням генератора білого шуму, з частотою генератора, рівною частоті тактування.

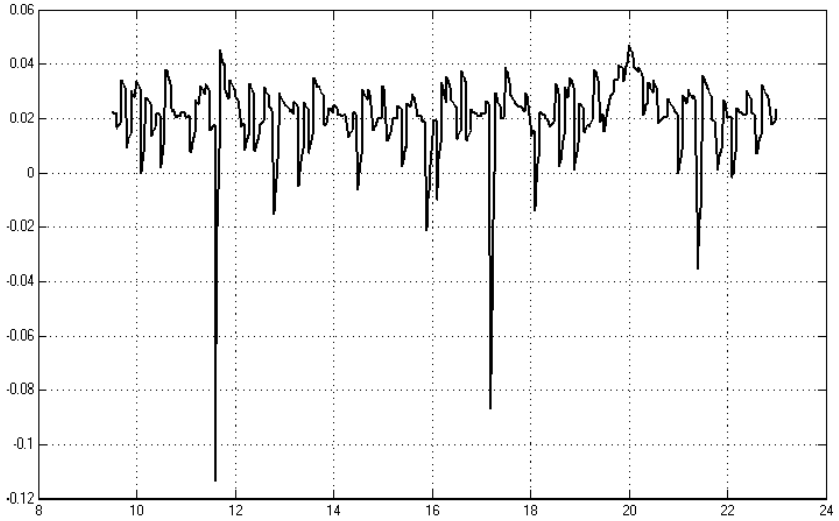


Рис.4.6. Струм споживання моделі мікроконтролера з регульованим фільтром з використанням генератора білого шуму, з частотою генератора, вдесятеро більшою частоти тактування.

Для того, щоб дослідити вплив типу розподілу генератора шуму, а також частоти шуму, було виміряно коефіцієнти взаємної кореляції між струмом споживання без маскування та з маскуванням за допомогою регульованого фільтру в залежності від типу генератора шуму та від його частоти. Результати вимірювання занесені в таблицю 4.1.

Таблиця 4.1.

Тт/Тг	Гаусівский	Рівномірний	Райсівський	Релеївський	Білий
1	0,6001	0,598	0,5923	0,4937	0,659
0,5	0,5974	0,4854	0,5039	0,5448	0,6295
0,2	0,5076	0,4805	0,4658	0,4912	0,4761
0,1	0,4495	0,4034	0,4147	0,4266	0,2429

В якості порівнюваних типів генераторів шуму використовувалися генератори, які дають гаусівский, рівномірний, райсівський, релеївський, білий шум відповідно. Співвідношення між періодом тактування мікроконтролера (Тт) та періодом відповідного генератора (Тг) змінювалася від 1 до 0,1.

Чим менший коефіцієнт кореляції, тим менш схожі між собою дані струми споживання, і тим краще регульований фільтр. Відповідно, дані залежності відображаються у вигляді сімейства (рис.4.7)

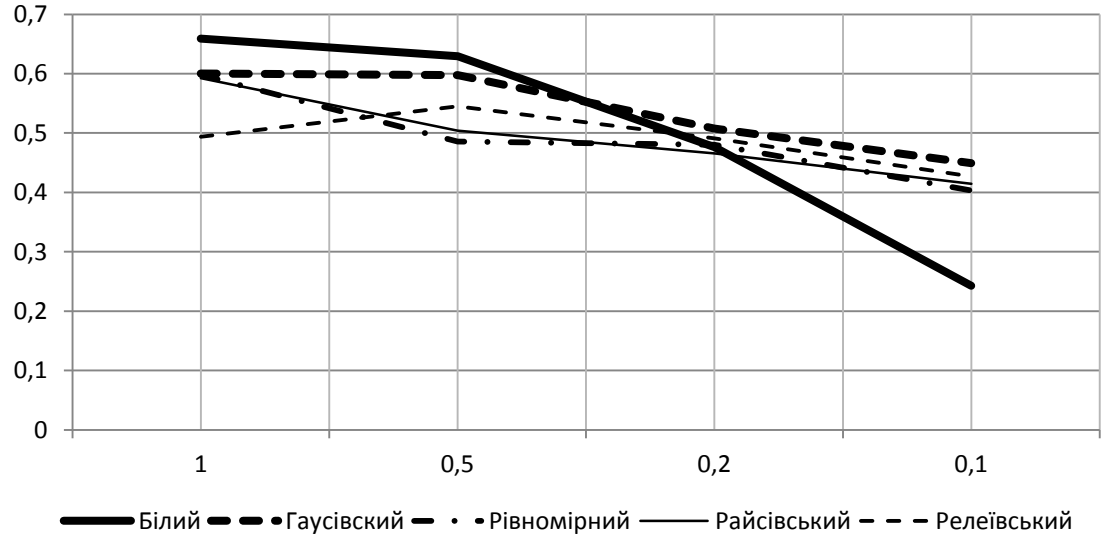


Рис.4.7. Залежності коефіцієнта взаємної кореляції від типу шуму та від співвідношення Тт/Тг

Із наведених вимірювань видно, що зі збільшенням частоти генератора, коефіцієнт взаємної кореляції зменшується для всіх типів шуму. Зменшення коефіцієнта взаємної кореляції означає краще маскування струму мікроконтролера. Для отримання прийнятного коефіцієнта взаємної кореляції менше 0,5 частота генератора шуму повинна бути щонайменше у 10 разів вищою, за частоту тактування мікроконтролера.

4.2. Регульований фільтр живлення мікроконтролера на основі змінного конденсатора (Регульований фільтр 1)

На рис.4.2 зображено модель регульованого фільтру із змінним конденсатором. Існує відомий пристрій [71], в якому паралельно до виводів джерела живлення підключений змінний конденсатор з електронним керуванням. Такі змінні конденсатори можуть бути реалізовані усередині мікропроцесорного кристала. Приклад виконання змінного конденсатора наведений у [72]. Ємність змінного конденсатора варіюється за допомогою схеми керування. Схема керування у відомому пристрої відслідковує значення напруги на шині живлення, і у випадку зменшення напруги живлення, зменшує ємність змінного конденсатора, а у випадку збільшення напруги живлення, збільшує ємність змінного конденсатора. Це дозволяє підтримувати напругу живлення обчислювального модуля відносно постійною, а отже, ускладнює аналіз струму споживання мікропроцесорної системи. Недоліком наведеного пристрою є наявність інерційного зворотного зв’язку у схемі керування, що дозволяє проходження струму живлення обчислювального модуля у ланцюг джерела живлення, а отже існує можливість аналізу струму живлення для визначення захищеної інформації.

Якщо до входу схеми керування змінним конденсатором підключити генератор шуму, такий, щоб частота генератора шуму була б вдесятеро

більшою за тактову частоту обчислювального модуля, можна підвищити захищеність мікропроцесорної системи від зчитування інформації за струмом споживання. Так як вхід схеми керування змінним конденсатором підключено до генератора шуму, то зміна ємності конденсатора буде відбуватися випадково, а отже до струму споживання будуть вноситись завади у вигляді випадкових стрибків струму. Оскільки частота генератора шуму є більшою за тактову частоту обчислювального модуля, ці випадкові стрибки струму практично неможливо відокремити від реального струму споживання обчислювального модуля.

На рис.4.8, зображено структурну схему регульованого фільтру з маскуванням інформаційних сигналів на основі змінного конденсатора.

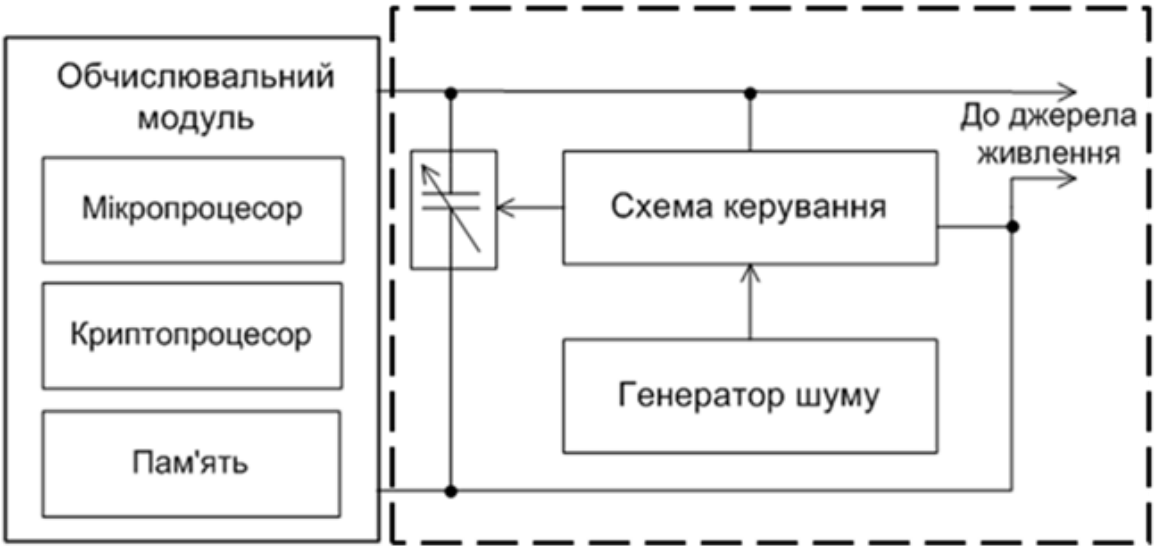


Рис 4.8. Запропонований регульований фільтр з маскуванням інформаційних сигналів на основі змінного конденсатора.

Регульований фільтр має обчислювальний модуль, який може містити у своєму складі мікропроцесор, мікроконтролер, оперативну пам'ять, постійну пам'ять, або будь-які інші цифрові пристрої. Паралельно до виводів живлення обчислювального модуля підключено конденсатор змінної ємності, виводи керування яким підключено до виходу схеми керування, до входу якої підключено генератор шуму.

Приклад реалізації генератора шуму та схему керування наведено на рис.4.9. За основу було обрано генератор шуму по документації га на елементах Maxim Integrated [73]. Джерелом шуму в генераторі шуму є зворотно зміщений стабілітрон VD1, за допомогою резистора R1 задається змішення стабілітрона. Шумовий сигнал з генератора шуму подається на вхід схеми керування, де поступає через роздільний конденсатор С1 на послідовно включені операційні підсилювачі DA1, DA2. З виходу операційних підсилювачів підсилений шумовий сигнал поступає на роздільний конденсатор С2, що використовується для вилучення постійної складової з шумового сигналу. З виходу схеми керування підсилений шумовий сигнал подається на керування змінним конденсатором. Живлення схеми відбувається через виводи VCC та GND.

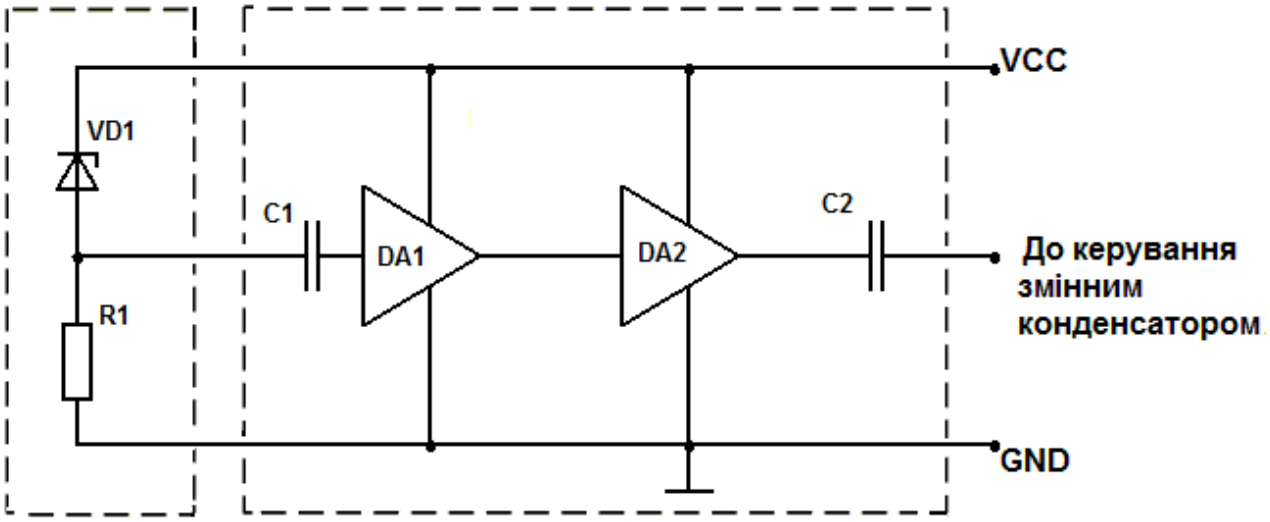


Рис 4.9. Приклад реалізації генератора шуму.

4.2. Регульований фільтр живлення мікроконтролера на основі блоку ключів (Регульований фільтр 2)

Пристрій зі схемою захисту від аналізу струму споживання (рис.1.10), містить центральний процесор, генератор випадкових чисел (ГВЧ), інтерфейс зв’язку, генератор випадкових станів (ГВС), блок ключів та пам’ять мікроконтролера. Недоліком вищевказаного пристрою є необхідність

використання складних ГВЧ на основі теплових квантових ефектів та необхідність певного їх розміщення на кристалі для забезпечення неідентичного функціонування. При цьому степінь захищеності суттєво залежить від складності, а значить і розміру ГВЧ.

В основу розробки запропонованого регульованого фільтру поставлено задачу удосконалення інтегрального пристрою зі схемою захисту від аналізу струму споживання, шляхом того, що додано блок керування ГВС. Вхід блоку керування ГВЧ з'єднаний з пам'яттю мікроконтролера, а вихід з ГВС, що забезпечує спрощення топології пристрою та покращення його характеристик.

Поставлена задача вирішується тим, що в мікроконтролер з системою захисту від атак за струмом споживання, додатково вводиться блок керування генератором випадкових станів, вхід якого з'єднаний з пам'яттю.

Розроблений пристрій містить інтерфейс зв'язку, що забезпечує обмін даними центрального процесора з зовнішніми пристроями та пам'ять мікроконтролера, в якій зберігаються конфіденційні дані. Для криптографічної стійкості застосовані ГВЧ, який з'єднаний з центральним процесором для забезпечення кодування даних, що передаються на зовні мікроконтролера. Система захисту містить блок ключів, приєднаних до живлення мікроконтролера, що керуються від ГВС і забезпечують додатковий струм споживання. На вхід ГВС подаються сигнали від блока керування ГВС, з'єднаного з пам'яттю мікроконтролера [74] для забезпечення передачі даних до, що збільшує період ГВС.

Пристрій для захисту мікроконтролера від атак за струмом споживання працює наступним чином. Через інтерфейс зв'язку центральний процесор отримує інструкції для виконання певних дій з даними пам'яті. Результат виконання обробки даних центральним процесором кодується завдяки ГВЧ працюючого не ідентично ГВС та передається через інтерфейс зв'язку до зовнішніх пристроїв. Живлення здійснюється через систему захисту від зчитування струму споживання. Центральний процесор виводами живлення

приєднаний до блоку ключів, ввімкнення різної кількості яких спотворює струм споживання пристрою в цілому. Керування блоком ключів здійснюється від генератора випадкових станів.

Структурна схема регульованого фільтру наведена на рис.4.10.

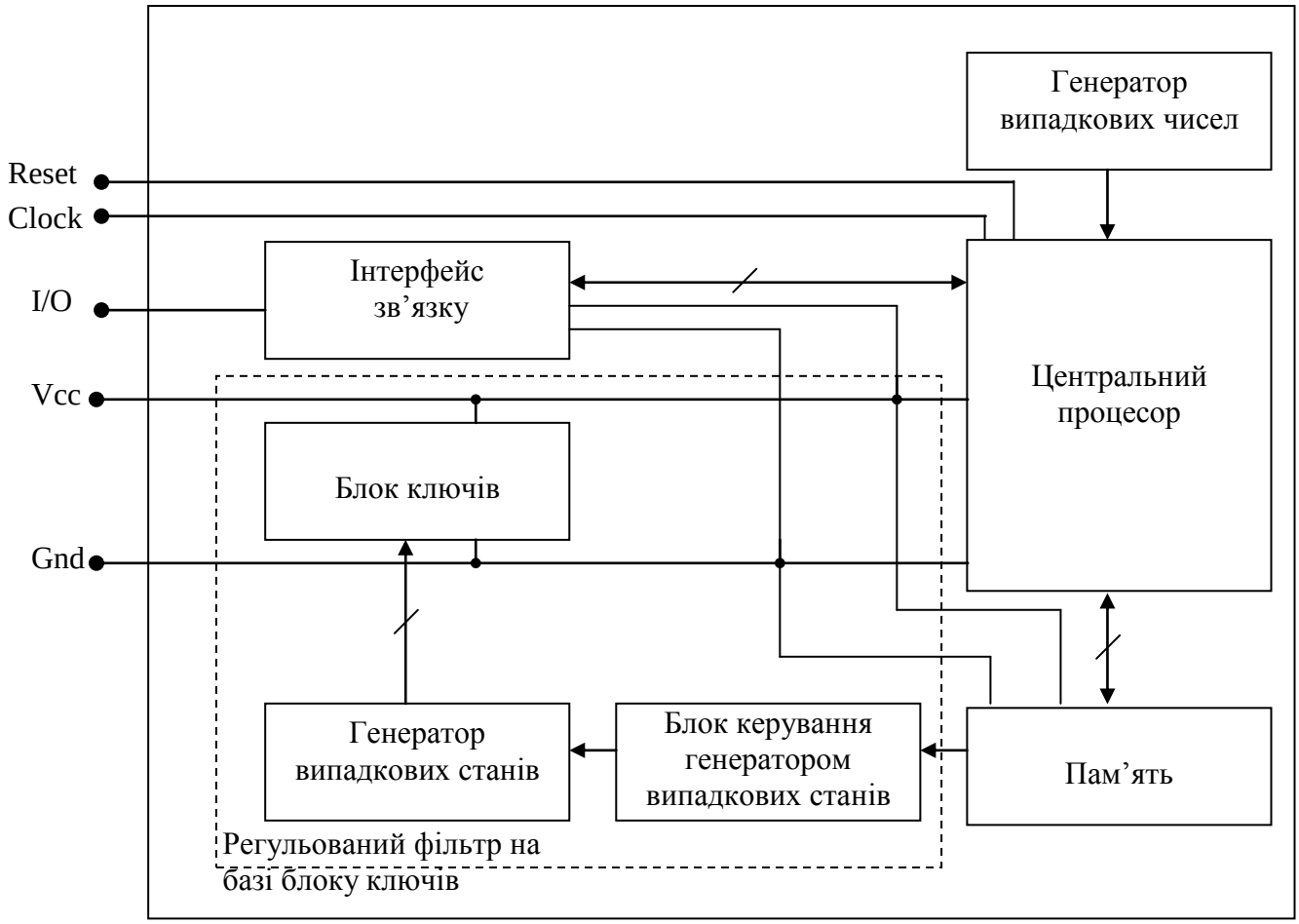
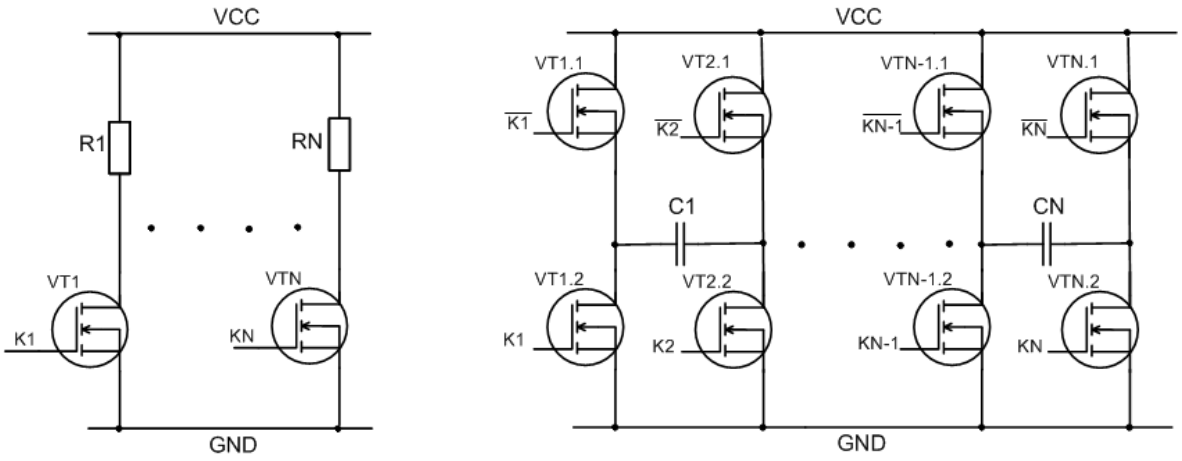


Рис.4.10. Структура регульованого фільтра на основі блоку ключів

Блок ключів складається з декількох комірок, кожна з яких містить ключ на МДН-транзисторі та навантаження. На рис.4.11.а зображена схема комірок блоку ключів, що підключають активне навантаження для завдання струму споживання, а на рис.4.11.б – схема комірок, що підключають ємнісне навантаження для створення перехідних процесів у струмі споживання.



а) б)

Рис.4.11. Схеми комірок блоку ключів

Завдяки блоку керування генератором випадкових станів, досягається внесення додаткових флуктуацій в генеровані ГВС випадкові числа, а отже стани блока ключів, та збільшення періоду флуктуацій.

Перевагою такого регульованого фільтру живлення є те, що він не містить конденсаторів великої ємності, а отже, займає меншу площу кристалу. Дану систему можна включати тільки по команді, що спричиняє економне енергоспоживання при незмінній захищеності. Крім того, є можливість програмно змінювати алгоритм генерації випадкових станів, що дає покращення захисту в наступних версіях системи.

4.3. Регульований фільтр живлення на основі допоміжного процесорного ядра (Регульований фільтр 3)

З огляду на наведені недоліки існуючих систем захисту мікроконтролерів від зчитування за струмом споживання розроблена нова система захисту, в якій запропоновано покращення характеристик захищеності від зчитування за струмом споживання.

Систему захисту виконано на основі додаткового процесорного ядра, яке виконує команди з пам’яті мікроконтролера. Це дозволяє замінити два блоки:

аналоговий ГВЧ та цифровий блок ключів одним цифровим пристроєм та виключити з топографії мікроконтролера аналогову схему ГВЧ. Додаткове ядро є програмованим пристроєм, і тому дозволяє використовувати алгоритми ГВЧ різної складності в залежності від потрібного ступеня захищеності, що дає можливість оптимізувати швидкодію мікроконтролера. Зв'язок основного ядра з допоміжним дозволяє використовувати динамічні дані, якими оперує мікроконтролер та використовувати їх в алгоритмі ГВЧ, що значно поліпшує характеристики алгоритму. Так, використання лише одного байту динамічних даних дає можливість збільшити період програмного алгоритму ГВЧ в $2^8 = 256$ разів, або використовувати більш прості швидкодіючі алгоритми ГВЧ, а отже і більшу тактову частоту центрального процесора (ЦП).

На рис.4.12 зображено структурну схему мікроконтролера із запропонованим регульованим фільтром живлення.

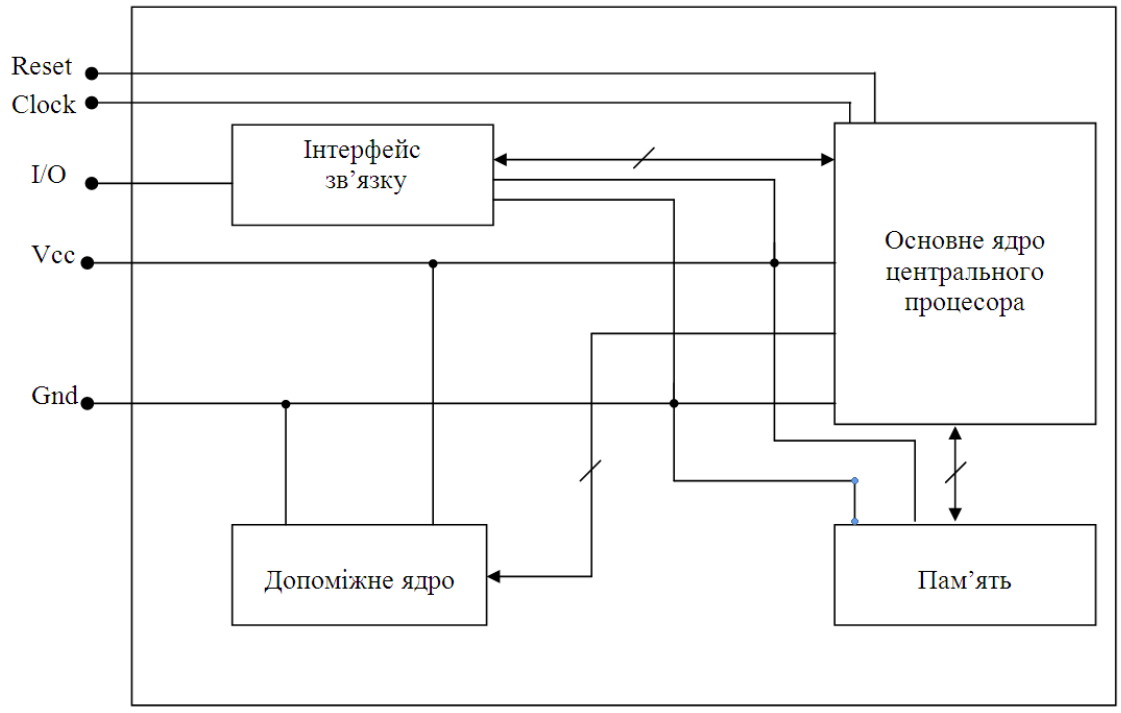


Рис.4.12. Мікроконтролер із запропонованим регульованим фільтром живлення

Мікроконтролер містить інтерфейс зв’язку, що забезпечує обмін даними ЦП із зовнішніми пристроями, та пам’ять, в якій зберігаються конфіденційні дані. Зв’язок мікроконтролера із зовнішніми пристроями здійснюється через порти інтерфейсу зв’язку „I/O”, сигнали синхронізації „clock”, „reset”, виводи живлення „Vcc” і „Gnd”.

Через інтерфейс зв’язку ЦП отримує інструкції для виконання певних дій щодо оперування даними. Результат виконання повертається через інтерфейс зв’язку до зовнішніх пристроїв. До виводів живлення мікроконтролера „Vcc” і „Gnd” паралельно під’єднано допоміжне процесорне ядро, що виконує функцію системи захисту від аналізу струму споживання. Використовуючи дані з пам’яті, додаткове ядро виконує команди, забезпечуючи при цьому внесення додаткових флуктуацій у струм споживання мікроконтролера в цілому. За рахунок зв’язку між основним та допоміжним ядром та пам’яттю досягається також вибір „зерна” (числа, що є основою програмних алгоритмів ГВЧ) алгоритму генерації ГВЧ з динамічно змінних даних пам’яті, що можуть змінюватися під час обчислень, суттєво збільшуючи період алгоритму, а отже, і ступінь захищеності мікроконтролера.

Реалізація додаткового захисного мікроконтролера може бути різною, в залежності від наявних ресурсів. Одним з варіантів реалізації може бути підключення зовнішнього мікроконтролера до шини живлення основного. Однак у даному випадку необхідно передбачити неможливість фізичного відключення захисного мікроконтролера. Це можливо досягти при використанні спеціальних корпусів із захисними пристроями, що знищують секретну інформацію при відкритті корпуса. Можливо також розподіляти основний алгоритм програми між основним та допоміжним ядром, і налаштувати обмін інформацією між ними, а у разі несанкціонованого доступу забезпечити знищення програмного пакету. Іншим варіантом виконання запропонованої системи захисту може бути використання сучасних двохядерних мікрокопроцесорів та мікроконтролерів [75]. При цьому одне ядро

буде використовуватись для виконання корисної програми, а інше ядро буде задіяне для виконання програми захисту. Ще одним з можливих варіантів реалізації системи захисту при реалізації пристрою на платформі ASIC є реалізація захисного мікроконтролера за допомогою вбудованої PLD-області. Слід також зазначити, що сьогодні можливе виготовлення фірмами-виробниками мікроконтролерів на замовлення, з необхідною конфігурацією вбудованих пристроїв та програмним забезпеченням. Тому, у пристроях із захистом інформації, коли собівартість даного пристрою відходить на другий план перед його захищеністю, реалізація додаткового захисного ядра процесора на одному кристалі є цілком можливою і оправданою. Ще одним можливим варіантом реалізації системи захисту від атак за струмом споживання є використання одноядерного мікроконтролера з програмою, що випадково маніпулює внутрішніми ресурсами – наприклад, включення та відключення АЦП, компаратора, підтягуючих резисторів на портах. Це призводить до введення додаткового випадкового шуму до струму споживання мікроконтролера, а отже значно ускладнює аналіз струму споживання.

4.4. Регульований фільтр живлення з вимірюванням струму у реальному масштабі часу (Регульований фільтр 4)

Регульований фільтр з маскування струму споживання із детектуванням послідовностей команд у реальному масштабі часу наведено на рис.4.13.

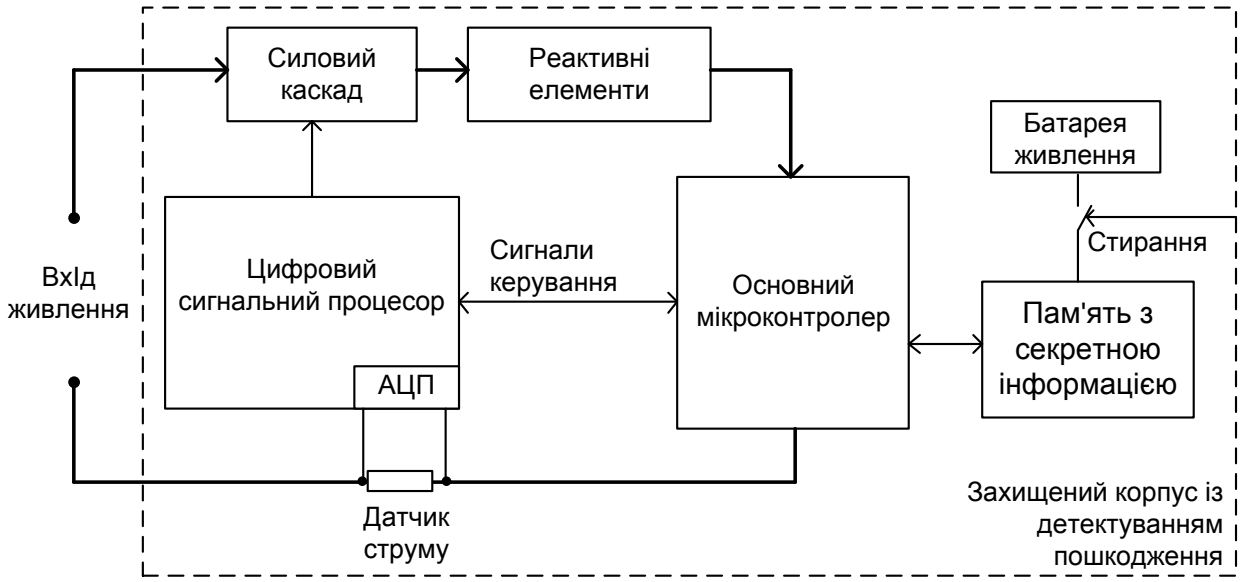


Рис.4.13. Структурна схема регульованого фільтру з детектуванням струму споживання у реальному масштабі часу

Регульований фільтр містить силовий каскад, реактивні елементи, що разом забезпечують спотворення струму споживання заданої форми та систему керування на базі цифрового сигнального процесора DSP. До процесора підключено сигнал зворотного зв’яжу по струму основного мікроконтролера, що перетворюється у двійковий код за допомогою АЦП, та оброблюються програмним забезпеченням. Для вимірювання струму споживання використовується датчик струму на неіндуктивному резисторі. Використання такого резистора дозволяє зменшити спотворення, викликані паразитною індуктивністю резистора та покращити точність вимірювання струму споживання. Також, введено цифровий зв'язок між основним мікроконтролером та процесором системи керування для керування режимами роботи джерела живлення. Наприклад, можливе включення додаткового спотворення струму споживання у місцях основного алгоритму, що працюють з секретними даними. Секретні дані, які необхідно захистити, містяться у енергозалежній пам’яті, яка може бути як у вигляді окремої мікросхеми, так і вбудована у мікроконтролер. При реалізації розроблюваного джерела живлення природно постає питання: як захиститися від випадку, коли зловмисник підключає основний мікроконтролер в обхід від захисного джерела живлення? Адже в такому випадку можливо

зчитування струму споживання основного контролера безпосередньо. Цю проблему пропонується вирішувати одним зі наведених нижче шляхів:

1) Підключення джерела живлення із захистом в одному корпусі з мікроконтролером. Така реалізація цілком можлива при використанні багатоядерних мікропроцесорів, або при реалізації системи керування на ASIC-платформі, виконати додатковий мікроконтролер можна на вбудованій PLD-області.

2) Виконання джерела живлення та основного мікроконтролера усередині корпуса, що відслідковує несанкціоноване відкриття, та стирає всю конфіденційну інформацію у енергонезалежній пам’яті. Тим самим зловмисник втрачає доступ до конфіденційної інформації. Такий принцип захисту реалізований, наприклад у криптопроцесорах IBM 4758 [76]. Криптопроцесор містить енергозалежну пам'ять, що містить секретний ключ шифрування, і живиться від вбудованої в модуль батареї, при відключеному зовнішньому живленні. При відкриванні корпуса, захисна мембрана вкриває плату криптомодуля, пошкоджується. Захисна мембрана складається з двох шарів мідних провідників, між якими прокладено тонкий шар діелектрику. При замиканні провідників між собою унаслідок пошкодження шару діелектрику, живлення з енергозалежної пам’яті секретного ключа знімається, тим самим відбувається стирання секретного ключа.

На схемі (рис.4.13) не зображено інші додаткові мікросхеми пам’яті, інтерфейси введення-виведення, оскільки вони не є основними при досягненні цілей захисту мікропроцесорної системи за струмом споживання.

При розробці алгоритму програмного забезпечення слід мати на увазі, те що ресурси захисного процесора обмежені, і тому необхідно якомога більше прискорити алгоритм [77]. Зокрема пропонується деякі операціє виконувати паралельно, при наявності такої можливості, або квазі-паралельно, тобто переключаючись між гілками алгоритму з почерговим обчисленням результатів.

Основні гілки виконання алгоритму (рис.4.14) наступні:

1) Зчитування поточного струму споживання на ¼ періоду виконання. Отримані дані з АЦП струму споживання поступають на блок визначення інтегральної характеристики у полярних координатах та паралельно на блок визначення коефіцієнтів СКІ перетворення в полярних координатах. Етапи перетворення даних АЦП ілюструє рис. 4.15 Дані струму споживання переводяться з декартової системи координат у полярну. Потім обчислюється інтегральна характеристика та коефіцієнти СКІ перетворення

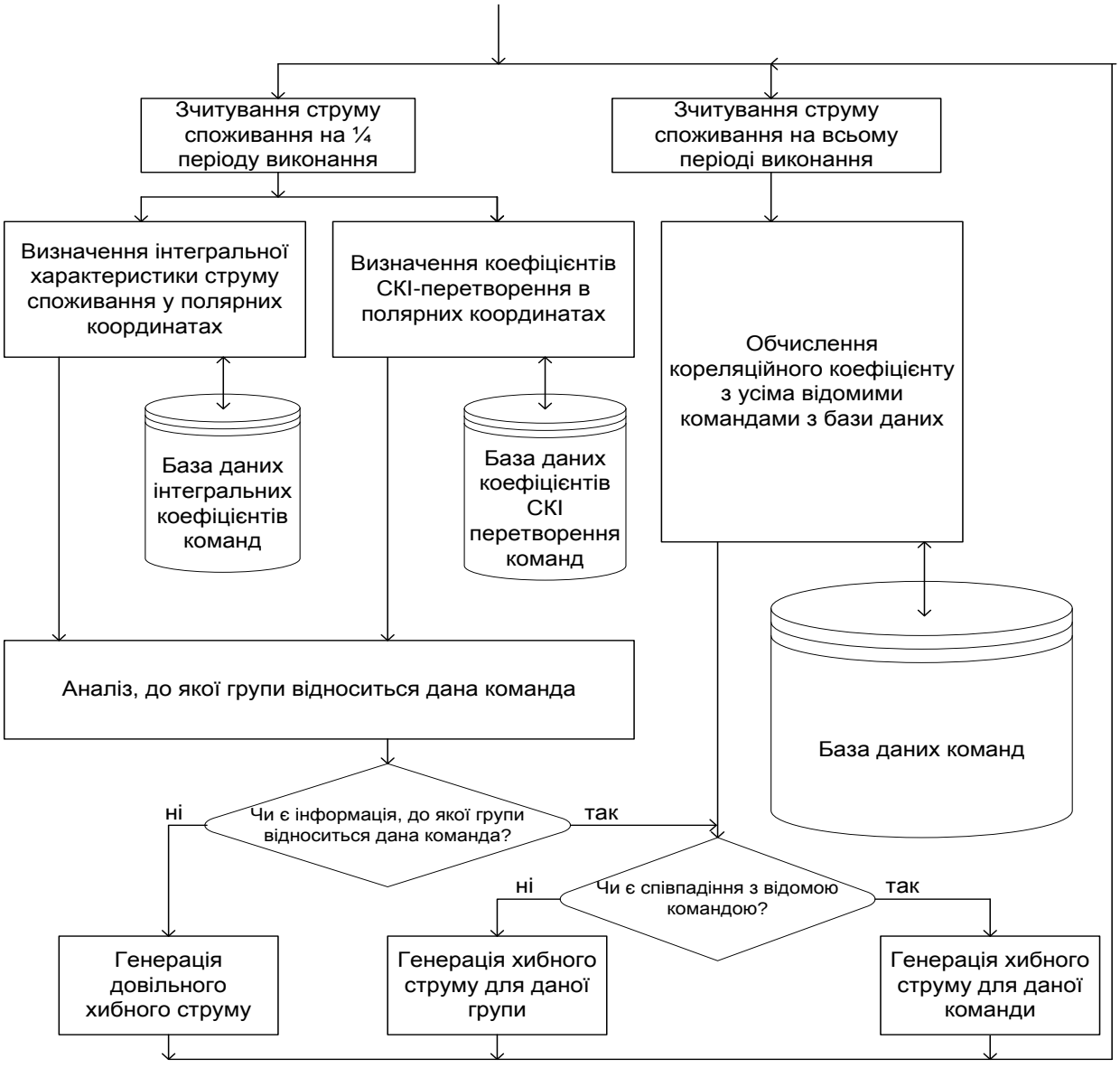


Рис.4.14. Алгоритм роботи системи керування регульованим фільтром у реальному масштабі часу

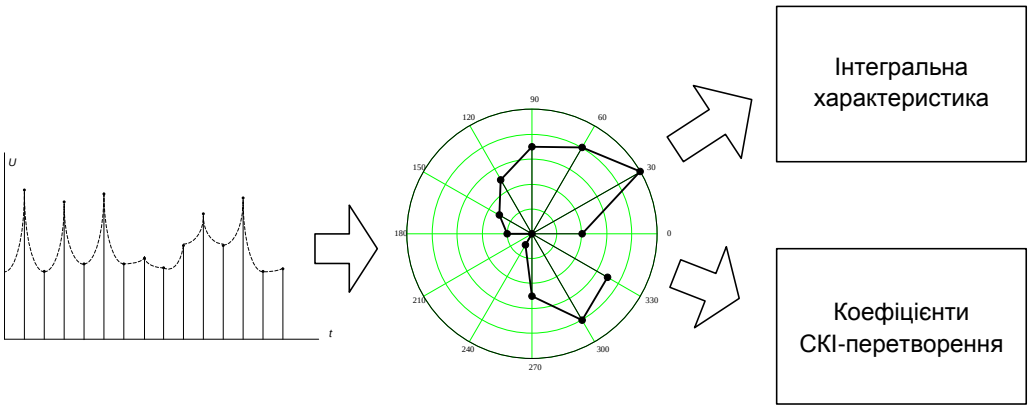


Рис.4.15. Перетворення даних струму споживання та їх обробка

Порівнюючи коефіцієнти, що отримані після обчислень у реальному часі з коефіцієнтами, що задані заздалегідь у базі даних, програма робить висновок про приналежність команди до певної групи. Однією з особливостей пошуку у базі даних є те, що числа у базі даних відсортовані, що значно прискорює пошук. Також, прискоренню пошуку та зменшенню обсягу пам‘яті сприяє є той факт, що для кожної команди записується тільки одне число – його інтегральна характеристика.

2) Зчитування струму споживання на усьому періоді виконання команди або послідовності команд та пошук у базі даних команд шляхом обчислення коефіцієнтів кореляції кривих струмів невідомої команди відомими струмами усіх команд бази даних. Ця база даних є об’ємною, оскільки містить інформацію про усі відліки кривих команд мікроконтролера. також пошук у базі відбувається шляхом порівняння з кожною строчкою таблиці поточної команди, що також не сприяє швидкодії алгоритму.

Оскільки наведені гілки алгоритму виконуються паралельно, через деякий час можна зробити висновок про те, до якої групи команд відноситься дана команда, та яка це команда, при чому дані про групу команди будуть отримані швидше. На основі цих даних пропонується наступний алгоритм прийняття рішення: якщо невідомо, до якої групи відноситься дана команда, генерується довільний струм споживання, якщо можна визначити групу команд, та немає інформації про те, яка це конкретно команда, або на останнє на вистачає часу,

то генерується хибна команда для даної групи. Якщо можна швидко визначити яка команда конкретно зараз виконується, генерується хибна команда для даної реальної команди.

Наведений алгоритм дозволяє, з одного боку, швидко генерувати хибні значення струмів команди, а з іншого боку забезпечує якомога швидший пошук реальних команд та відповідних струмів спотворення.

Таким чином, застосування генератора шуму при реалізації системи керування регульованим фільтром живлення дозволяє в цілому збільшити ефективність захисту за струмом споживання. Із збільшенням частоти генератора, коефіцієнт взаємної кореляції зменшується для всіх типів шуму.

РОЗДІЛ 5

ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ЗАПРОПОНОВАНИХ РЕГУЛЬОВАНИХ ФІЛЬТРІВ

5.1. Структурна схема експериментальної установки

Ефективність існуючих систем захисту з із запропонованими системами керування джерелами живлення розглянемо на прикладах систем із застосуванням:

1) вхідного фільтруючого конденсатора;
2) фільтрів зі змінними параметрами;
3) стабілізатора напруги;
4) програмної системи захисту з маніпуляцією внутрішніми ресурсами мікроконтролера.

Також для порівняння використуємо запропоновані системи на базі регульованих фільтрів джерел живлення на основі:

1) змінного конденсатора (Регульований фільтр 1);
2) блоку ключів (Регульований фільтр 2);
3) допоміжного процесорного ядра (Регульований фільтр 3);
4) вимірювання струму у реальному масштабі часу (Регульований фільтр 4).

Оскільки виконання наведених систем захисту потребують близької за структурою системи керування, то доцільно виконати її єдиним блоком з можливістю розподілення сигналів керування. Для цього використовуємо мікроконтролер з великою кількістю інтегрованих пристроїв, необхідних для реалізації програмного захисту.

Моделювання систем зі змінним конденсатором при роботі мікроконтролера на частоті порядку 1 МГц можна виконати лише з використанням комутованих електронними ключами конденсаторів. Відповідно

ці ж самі ключі можна використати для моделювання системи захисту на основі блоку ключів, де замість конденсаторів використовуються резистори, що обмежують струм через ключі при їхньому паралельному підключенні до виводів живлення.

Оцінити ефективність системи з застосуванням фільтрів зі змінними параметрами, що загалом може мати складну каскадну структуру, досить просто використовуючи один фільтр зі змінними параметрами, одиничний вплив якого дає змогу встановити необхідну їх кількість для досягнення потрібного впливу на струм споживання.

Необхідно також виконати роздільне живлення основного мікроконтролера із захистом та системи керування змодельованими системами захисту для уникнення явища взаємного накладання їхніх струмів споживання тим самим досягається підвищення чистоти експерименту.

Оскільки мікроконтролер з системою захисту повинен виконувати декілька алгоритмів зі схожими та відмінними частинами необхідно передбачити індикацію поточного алгоритму.

Виконуючи вимірювання неможливо уникнути впливу від електромагнітного випромінення мережі та вимірювальних приладів, а також інших завад, що присутні в навколишньому просторі. Для зменшення їхнього впливу систему захисту проектуємо з захисним екраном на зразок комірки Фарадея.

Підсумувавши наведені спрощення та вимоги, отримуємо структурну схему тестового макету (рис.5.1).

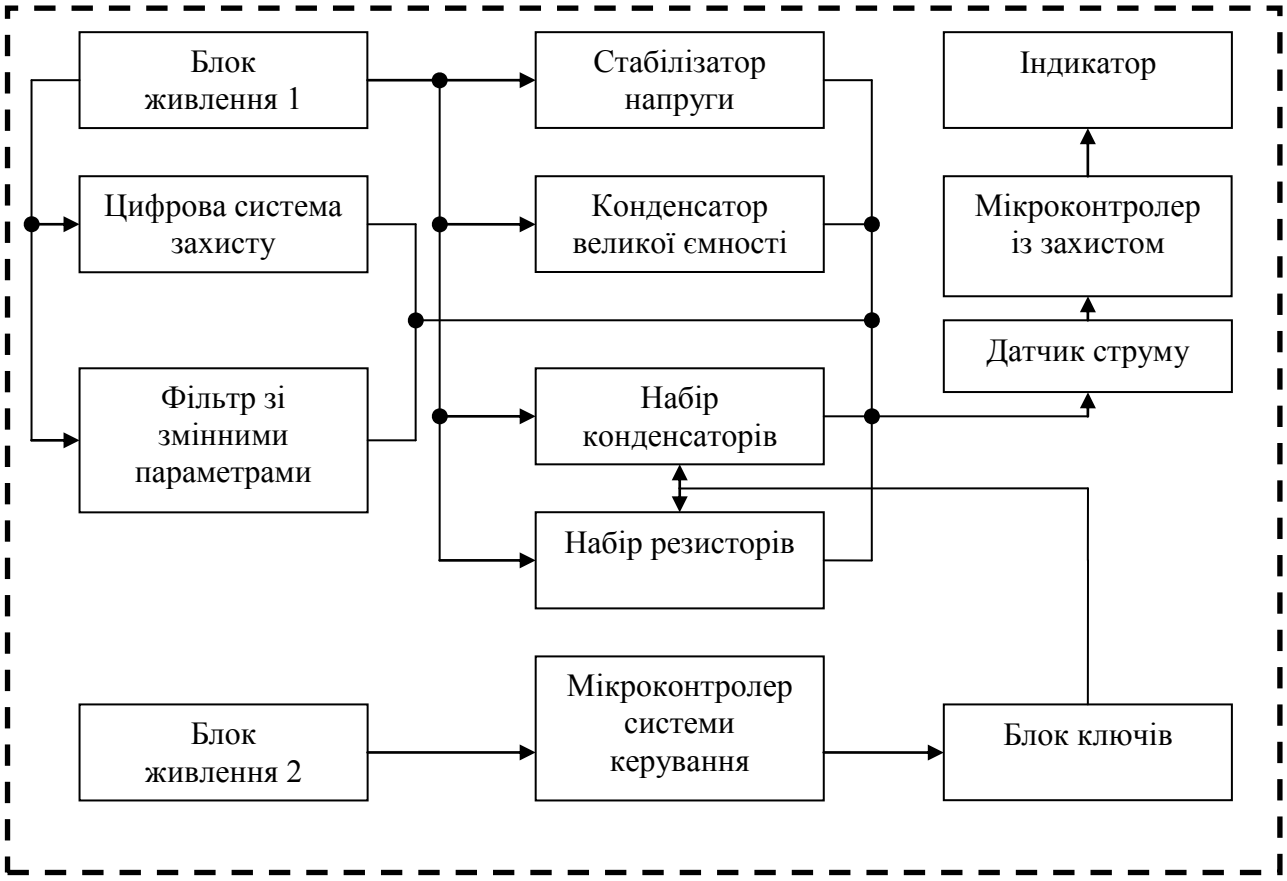


Рис.5.1. Структурна схема тестового макету моделювання семи регульованих фільтрів з маскуванням струму споживання

5.2. Принципова схема тестового макету

Взявши за основу структурну схему (рис.5.1) виконаємо на її основі схему електричну принципову тестового макету, замінивши блоки відповідними електронними компонентами (рис.5.2). Тестовий макет призначений для безпосереднього виконання вимірювань струмів споживання при використанні однієї з семи систем захисту від атак за струмом споживання (чотирьох відомих та перших трьох запропонованих регульованих фільтрів), та передбачає підключення всіх необхідних вимірювальних та допоміжних пристроїв.

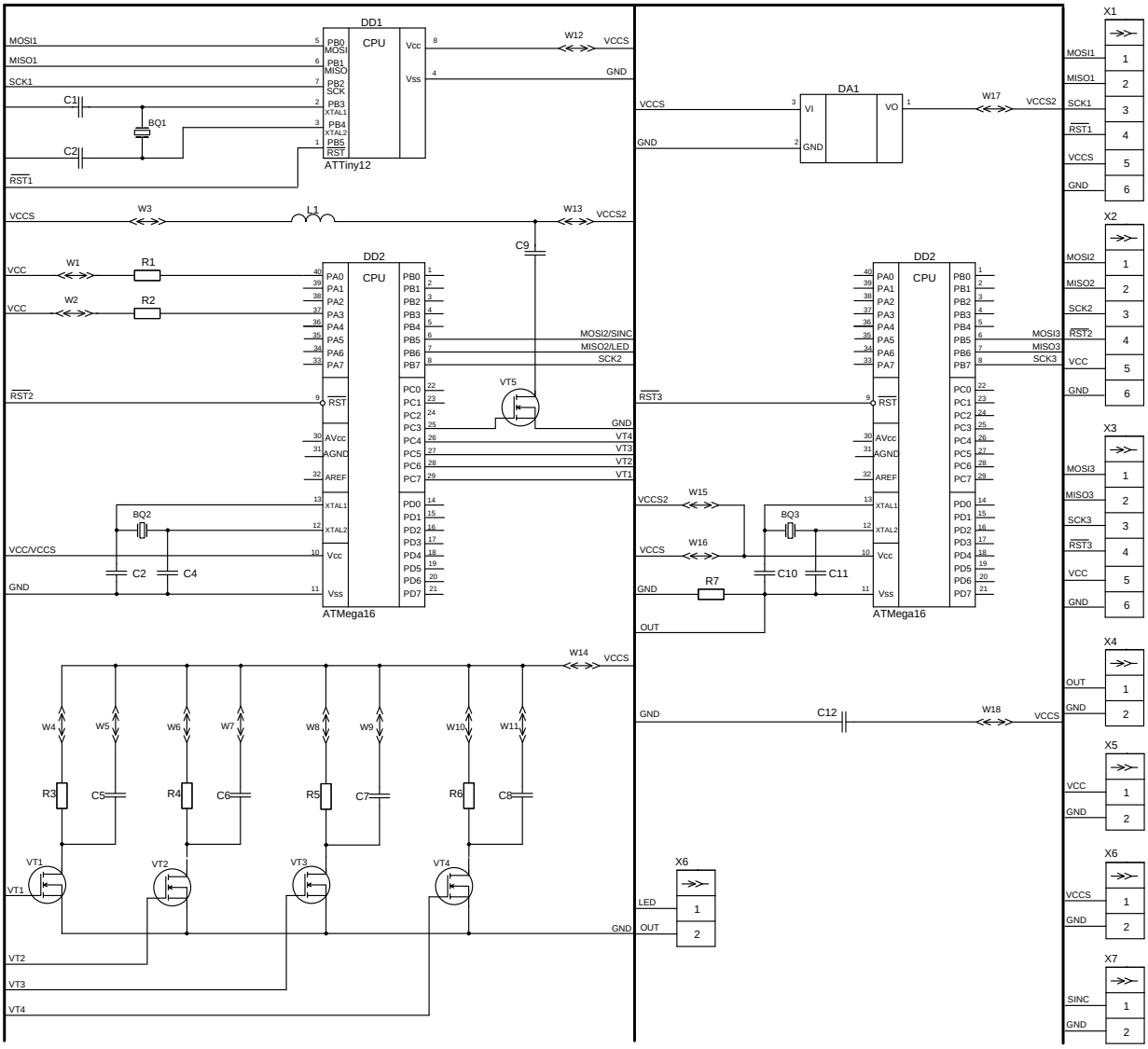


Рис.5.2. Схема електрична принципова тестового макету моделювання семи систем захисту від атак за струмом споживання.

Основою тестового макету є мікроконтролер AVR ATMega 16 [78] [79] [80], що підлягає захисту *DD3*. Він живиться від джерела живлення, приєднаного до роз'ємну *X6*. До цього самого джерела паралельно за допомогою перемичок *W1-W18* підключаються різноманітні системи захисту. Для підключення осцилографа і зняття струму споживання призначені датчик струму виконаний на резисторі *R7* та роз'єм *X4*.

Мікроконтролер *DD1* представляє собою модель розробленої цифрової системи захисту на основі допоміжного ядра, а мікроконтролер *DD2* виконує

функцію системи керування всіма іншими системами захисту. Вибір підпрограми системи керування здійснюється за допомогою перемичок *W1, W2* та обмежуючих струм резисторів *R1, R2*, які дають змогу реалізовувати до чотирьох різних розгалужень основної програми. Живлення системи керування здійснюється через роз'єм *X5*.

Робочі частоти мікроконтролерів задаються кварцовими резонаторами *BQ1-BQ3* з конденсаторами *C1-C4* та *C10, C11*, необхідними для коректної роботи на відповідній частоті. Функціонування кожного мікроконтролера від власного задаючого генератора на основі кварцового резонатора дає можливість проводити подальший аналіз впливу співвідношення частот на ефективність існуючих і запропонованої систем захисту.

Блок ключів реалізовано на МДН-транзисторах *VT1-VT4* підключених до порту С мікроконтролера *DD2*, що виконує їх переключення у відповідності до запропонованого алгоритму. Навантаженням транзисторів можуть слугувати конденсатори *C5-C8* або резистори *R3-R6*, включення яких визначається перемичками *W4-W11*.

Транзистор *VT5* разом з індуктивністю *L1* та конденсатором *C9* відповідають фільтру зі змінними параметрами на структурній схемі. Керування транзистором здійснюється за таким самим алгоритмом, що і для *VT1-VT4*, тому затвор *VT5* також приєднано до порту С мікроконтролера *DD2*.

Через перемички *W18* та *W17* можна відповідно забезпечити живлення з додатковим фільтруючим конденсатором *C12* або через стабілізатор напруги *DA1*.

Для забезпечення швидкої модифікації програмних кодів мікроконтролерів передбачено роз'єми послідовного програмування *X1-X3* для стандартного програматора.

Функціонування схеми здійснюється наступним чином. Спочатку необхідно обрати систему захисту, що моделюється, для чого виконати встановлення перемичок згідно табл. 5.1.

Таблиця 5.1

Наявність перемички / Система захисту	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18
Вхідний фільтруючий конденсатор	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	-	+
Фільтр зі змінними параметрами	+	-	+	-	-	-	-	-	-	-	-	-	+	-	+	-	-	-
Стабілізатор напруги	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	-	+	-
Система захисту на основі блоку ключів	+	-	-	+	-	+	-	+	-	+	-	-	-	+	-	+	-	-
Система захисту на основі змінної ємності	+	-	-	-	+	-	+	-	+	-	+	-	-	+	-	+	-	-
Маніпуляція внутрішніми ресурсами	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-	+	-	-
Система захисту на основі додаткового ядра	-	-	-	-	-	-	-	-	-	-	-	+	-	-	-	+	-	-

Після цього підключення живлення до роз’ємів *X5* та *X6* забезпечить роботу мікроконтролера із захистом через відповідну систему захисту. Сумарний струм споживання від джерела живлення, підключеного до *X6*, протікає через обрану систему захисту, мікроконтролер *DD3* та датчик струму *R7*. За рахунок виконання внутрішнього алгоритму кодування, що запрограмований в *DD3*, стум споживання буде постійно змінюватись у відповідності до миттєвої інтенсивності обчислень. В залежності від підключеної системи захисту відбувається згладжування струму споживання

або накладання на нього струму споживання, що визначається системою керування *DD2* чи мікроконтролером *DD1*.

5.3. Програмне забезпечення для цифрової обробки результатів експерименту

Тестовий макет містить три програмованих мікроконтролера, що потребують розробки відповідних програмних продуктів для реалізації алгоритмів тестових струмів, системи керування та системи захисту відповідно.

Програмне забезпечення основного мікроконтролера повинно задовольняти наступним вимогам:

1) циклічно виконувати одну підпрограму.

2) мати можливість відслідкувати часовий інтервал, на якому виконується дана підпрограма.

3) мати можливість виконувати декілька команд мікроконтролера з якомога меншим числом циклів перепрограмування.

4) мати можливість простої індикації номеру поточної виконуваної підпрограми для того, щоб при зніманні експериментальних даних була можливість визначити підпрограму, яка виконується в даний момент.

Основний мікроконтролер циклічно видає послідовності мікрокоманд, що підлягають аналізу та імпульси синхронізації, за якими можна визначити початок та кінець кожного алгоритму та забезпечити синхронізацію при записі та аналізі отриманих струмів споживання. Алгоритм його роботи наведено на рис.5.3.

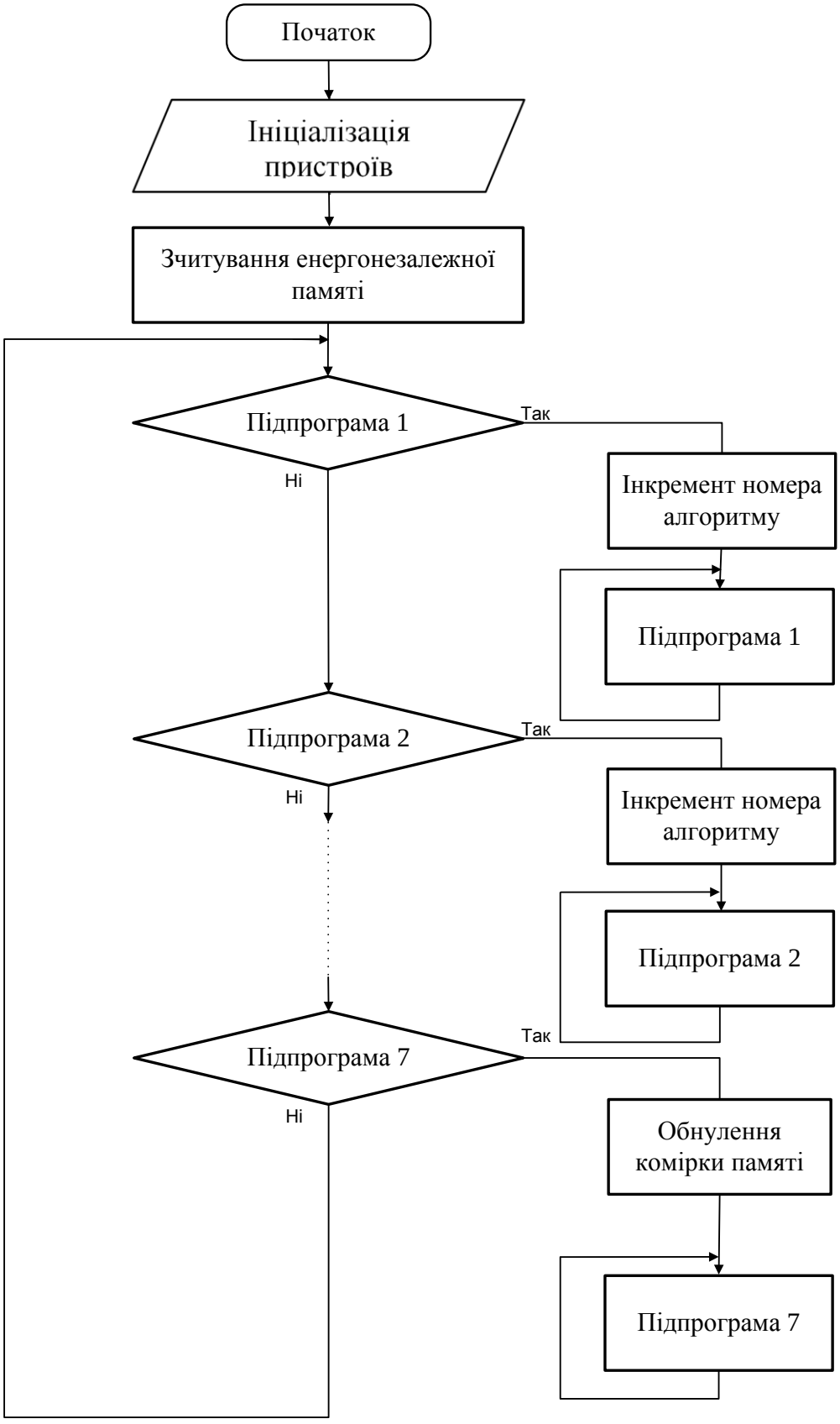


Рис.5.3. Алгоритм роботи мікроконтролера, що підлягає захисту.

Підпрограми 1-7 виконують наступні функції:

1) Підпрограма 1 «PORT» записує в усі порти вводу-виводу значення, які відповідають усім логічним нулям, а потім усім логічним одиницям. При переключенні портів з одного такого стану в інший спричиняє значні перепади у струмі споживання, оскільки потужні польові транзистори вихідних регістрів порта мають велику ємність

2) Підпрограма 2 «NOP» складається з 9 команд NOP і не виконує ніяких операцій з портами введення-виведення, з АЛП або пам’яттю, тому повинна мати найменші флуктуації струму споживання. Струм даної підпрограми можна порівнювати з іншими струмами та відмітити відмінність між ними.

3) Підпрограма 3 «SBI/CBI» встановлює та скидає лише один біт порту вводу-виводу командами SBI/CBI та призначена для вивчення впливу переключення одиничного розряду регістру порту на струм споживання мікрокотнтролера.

4) Підпрограма 4 «EOR/ROL» виконує циклічний зсув EOR/ROL бітів у регістрі, при чому регістр на початку заповнений так, щоб на кожному кроці виконання даної підпрограми змінювалося значення усіх бітів регістру. В цій підпрограмі не використовується лише операції з регістрами загального призначення, та не використовуються порти введення-виведення, що дає змогу дослідити вплив зміни значень в регістрах на струм споживання, оскільки операції з регістрами є основними операціями, які мікроконтролер виконує більшість часу у звичайних програмах.

5) Підпрограма 5 зчитує дані з енергонезалежної пам'яті EEPROM. Енергонезалежна пам'ять є складним пристроєм, саме тому звернення до неї повинно спричиняти помітні сплески струму споживання.

6) Підпрограма 6 «NOP+PORT» є комбінацією з наведених вище підпрограм 2 «NOP» та 1 «PORT», які виконуються одна за одною. Така комбінація була обрана для того, щоб на одній осцилограмі побачити відмінність між даними програмами та оцінити вплив конвеєра команд на струм споживання.

7) Підпрограма 7 «зчитування LPM і запис в SRAM» - зчитує байт з енергонезалежної пам’яті програм та записує його до статичної пам’яті даних. Призначення даної підпрограми в тому, щоб показати вплив вказаних операцій на струм споживання. Обидва виду пам’яті є досить важливими та потужними частинами мікроконтролера, тому мають спричиняти значні флуктуації струму споживання.

Кожна підпрограма тривалістю 15 машинних тактів формує імпульс синхронізації та певну послідовність мікрокоманд, розроблену таким чином, щоб зняті без систем захисту струми споживання мали різні за значенням коефіцієнти взаємної кореляції. Струми споживання деяких підпрограм дуже схожі, в той час як для інших вони суттєво відрізняються.

Індикація виконуваної підпрограми забезпечується за рахунок функції виводу відповідної кількості імпульсів на порт мікроконтролера до якого підключається світло діод. Після скидання мікроконтролера світлодіод спалахує таку кількість разів, що відповідає номеру виконуваної далі циклічної підпрограми.

Компіляцію виконано за допомогою програмного забезпечення AVR Studio 4.0.

Мікроконтролер системи керування забезпечує керування трьома модельованими системами захисту і реалізує систему захисту на онові маніпуляції внутрішніми ресурсами.

Відповідно до принципової схеми можливе переключення чотирьох програм без перепрограмування, однак алгоритм можна побудувати так, щоб використовувати лише одне розгалуження основної програми. Воно задається положенням перемичок *W1* та *W2*, які встановлюють на виводах порту А логічну одиницю.

Розроблена на мові програмування С [81] [82] програма скомпільована за допомогою ImageCraft 7 for AVR. Вона функціонує згідно наведеного на рис.5.4 алгоритму та має п’ять основних блоків:

1) Функції ініціалізації внутрішніх пристроїв.
2) Функція затримки.
3) Функція вибору підпрограми.
4) Підпрограму генерації 5-бітного випадкового числа.
5) Підпрограму маніпуляції одним з п’яти вбудованих ресурсів.

Використання програмної оболонки ImageCraft 7 for AVR дозволяє за рахунок утиліти Application Builder проводити автоматичну ініціалізацію внутрішніх ресурсів на потрібний режим та напряму звертатися до апаратних регістрів.

Функціонування програми згідно наведеного алгоритму відбувається за наступною послідовністю. Спочатку ініціалізуються всі внутрішні ресурси після чого вони переходять в режим очікування та не споживають динамічної складової струму. За рахунок апаратного переключення здійснюється розгалуження на два алгоритми.

Перший алгоритм використовує бібліотечну функцію генератора випадкових чисел rand(), що за рахунок відповідних обмежень та зсуву видає на порт мікроконтролера п’ять біт відповідних 5-розрядному випадковому числу. Маніпуляція цими бітами спричиняє виникнення імпульсів прямокутної форми на виводах PC3-PC7 порту C вводу-виводу, що призводить до випадкового переключення підключених транзисторів.

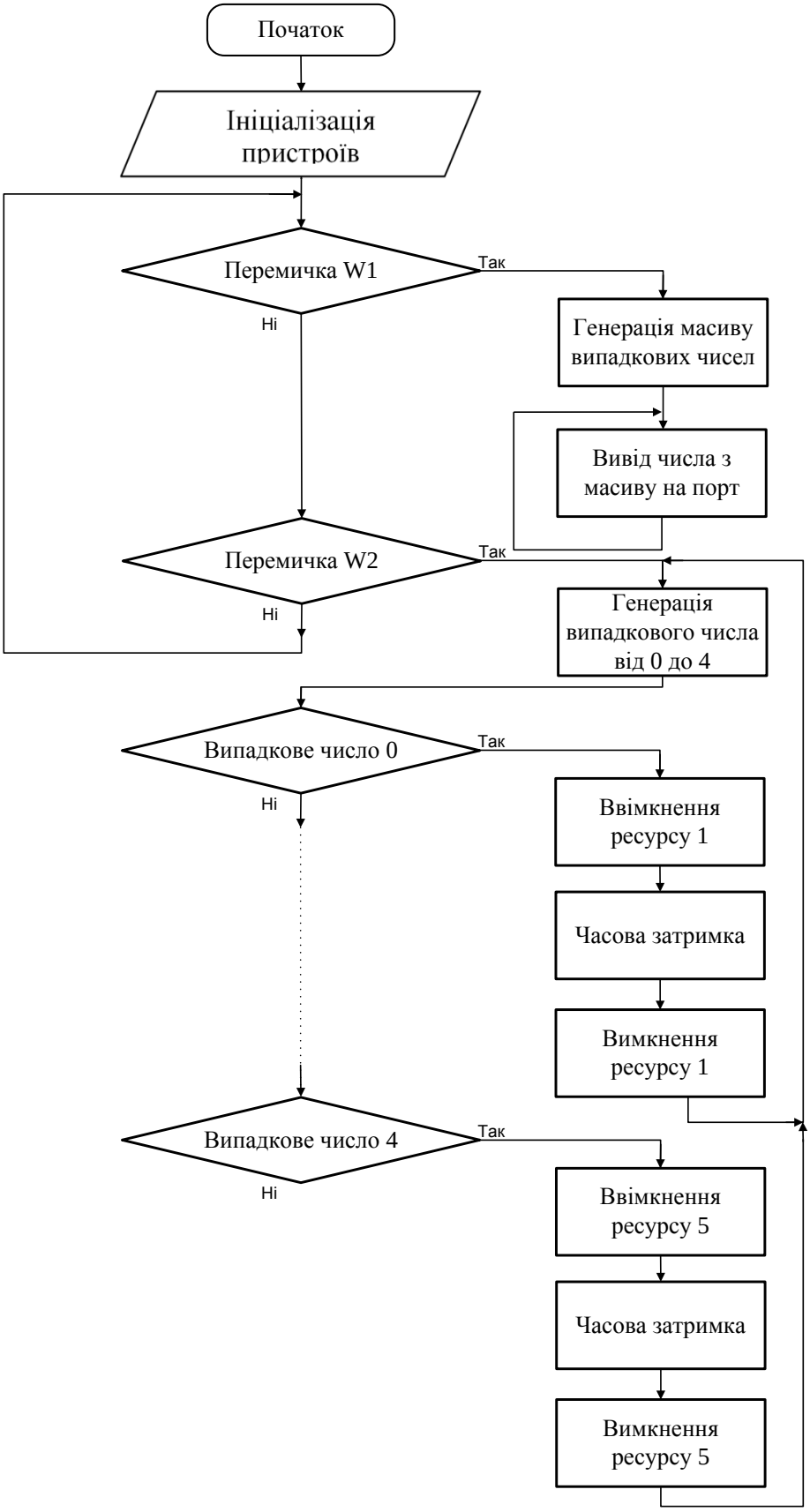


Рис.5.4. Алгоритм роботи мікроконтролера системи керування.

Для максимізації швидкодії генерація масиву випадкових чисел здійснюється лише на початку підпрограми, після чого в циклі виконується звернення до цього масиву. Це пояснюється тим, що функція rand() виконується впродовж багатьох машинних тактів, що збільшило б тривалість імпульсів керування, а отже зменшило робочу частоту систем захисту. Перезавантаження мікроконтролера призводить до генерації нового масиву випадкових чисел. За рахунок такої організації алгоритму зміна стану порту С відбувається кожних 4 такти.

Другий алгоритм базується на тій самій функції генерації випадкового числа, однак замість видачі його на порт вводу-виводу розгалужується на п’ять гілок маніпуляції різними мікроконтролерними ресурсами. Тривалість роботи кожного внутрішнього пристрою визначається функцією delay(), параметр якої задає кількість тактів протягом яких внутрішній ресурс буде задіяно на виконання типової задачі.

Програмне забезпечення для ATTiny12 забезпечує:

1) Генерацію шуму.

2) Використання мікрокоманд, аналогічних до підпрограм основного мікроконтролера.

3) Максимальна швидкодія.

4) Рівномірне використання комірок EEPROM.

5) Використання мінімального числа мікрокоманд.

Алгоритм, що моделює допоміжне ядро на основі ATTiny12 приведеного на рис.5.5.

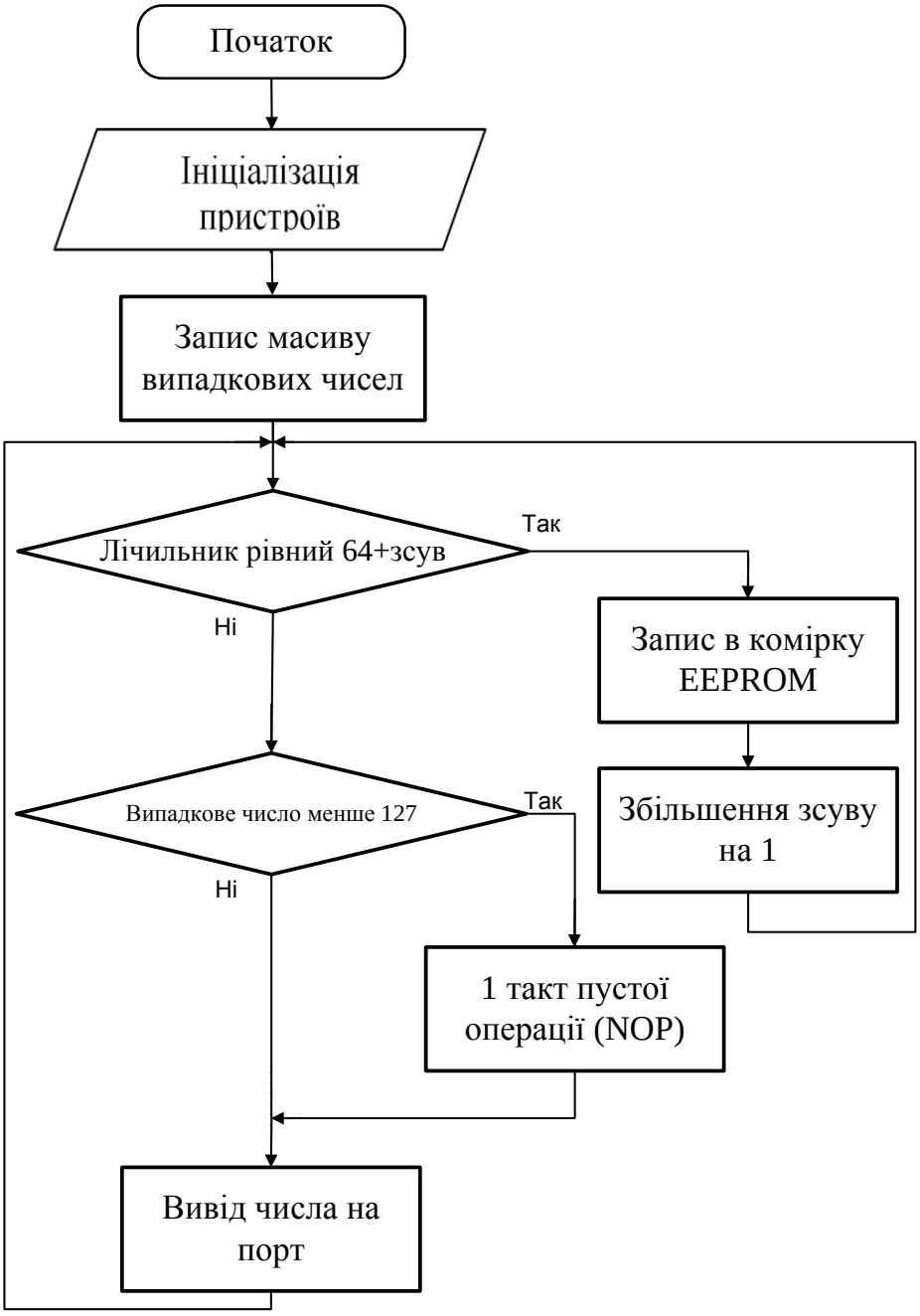


Рис.5.5. Алгоритм роботи
мікроконтролера моделюючого цифрову систему захисту.

Дуже незначний обсяг пам’яті та наявність лише регістрів загального призначення не дозволяють реалізувати складний генератор випадкових чисел, тому було вирішено використовувати готові 256-байтні масиви випадкових чисел, згенеровані комп’ютером.

Готова програма включає в себе ініціалізацію порту та енергонезалежної пам’яті, а також циклічний вивід на порт байту даних з записаної при програмуванні бази даних. Після перезапису виконання 64-х операцій здійснюється запис останнього байту в енергонезалежну пам’ять та зсув початкової адреси масиву випадкових чисел. Наявність в кожному циклі команди порівняння дозволяє отримати динамічну зміну періоду виконання циклу, тобто внести додатковий фактор випадковості.

Нижче наведено лістинг 5.1 розробленої в середовищі ImageCraft for Tiny програми:

Лістинг 5.1

```
// Target : T12
// Crystal: 4.0000Mhz
#include <iotiny12.h>
#include <macros.h>

//Масив випадкових чисел:
flash unsigned char RAND_LOOKUP[256] =
{
0x18, 0xC7, 0x0B, 0xF6, 0xFF, 0xB0, 0xAF, 0x2E,
0x2D, 0xF7, 0xB1, 0xE2, 0x20, 0x26, 0x45, 0x08,
0x53, 0xE6, 0x3D, 0xB3, 0xD4, 0x16, 0xD1, 0xCF,
0x34, 0x1B, 0x9F, 0x13, 0x76, 0xF0, 0x31, 0x4C,
0x9B, 0x3F, 0x50, 0x01, 0xF4, 0xBB, 0x6D, 0x75,
0xDA, 0x69, 0xB4, 0x60, 0x63, 0x3C, 0x3A, 0x12,
0xCA, 0xD2, 0x55, 0x6B, 0xFD, 0xB2, 0x7D, 0xD5,
0x57, 0x27, 0xE7, 0x82, 0xC9, 0x9D, 0x58, 0xF5,
0x5F, 0x4D, 0x38, 0xC5, 0xE4, 0x1D, 0xE3, 0xC1,
0x54, 0x89, 0x8B, 0x0E, 0xFB, 0x05, 0xA7, 0xF3,
0x1F, 0x62, 0x7E, 0x2B, 0xA6, 0x39, 0x0C, 0xE8,
0xAB, 0x11, 0x03, 0x36, 0x72, 0x49, 0xBF, 0x41,
0x59, 0xDD, 0xDB, 0x6A, 0x98, 0x30, 0xF1, 0x42,
0x52, 0x99, 0x61, 0xA4, 0xBD, 0xB5, 0x2F, 0x0F,
0x33, 0xF9, 0x28, 0x92, 0x77, 0x90, 0x3B, 0x04,
0xD9, 0x15, 0xB8, 0x44, 0xE9, 0x7F, 0xDF, 0xD3,
0x94, 0x81, 0x6E, 0x24, 0xAC, 0xFC, 0x2C, 0xA9,
0x46, 0xDE, 0x65, 0x19, 0x9E, 0x4F, 0xA1, 0x8C,
0xD6, 0xEE, 0xB7, 0xC6, 0x95, 0x85, 0x79, 0x96,
0x43, 0x29, 0x3E, 0x86, 0x5D, 0xFE, 0x51, 0x7A,
```

```
0xEC, 0x0A, 0xC3, 0x67, 0xC2, 0xEF, 0xE0, 0x2A,
0x6F, 0xB6, 0x91, 0x47, 0xD0, 0xA8, 0x74, 0x02,
0x87, 0x9C, 0xDC, 0xA2, 0xB9, 0xBA, 0x09, 0x14,
0xCC, 0x7B, 0xAE, 0x70, 0xD7, 0xC0, 0xE5, 0xD8,
0x83, 0x7C, 0x07, 0xF8, 0x71, 0x48, 0x0D, 0x78,
0x1C, 0xAA, 0x97, 0xA0, 0x23, 0xBC, 0x73, 0x1E,
0xF2, 0x93, 0x10, 0xC4, 0x37, 0x56, 0x8D, 0x22,
0x4E, 0xAD, 0x5E, 0x5C, 0x35, 0x9A, 0xED, 0xBE,
0x84, 0xA5, 0xA3, 0xCE, 0x5B, 0x66, 0x06, 0x1A,
0x5A, 0x4B, 0x8A, 0xCD, 0x64, 0xEA, 0x17, 0xC8,
0x8F, 0x8E, 0x88, 0xEB, 0x21, 0x68, 0x4A, 0x32,
0x6C, 0xCB, 0x25, 0x40, 0xFA, 0x80, 0xE1, 0x0E
};
```

//Функції ініціалізації:

```
void port_init(void)
{
 PORTB = 0x00;
 DDRB  = 0x00;
}
//call this routine to initialize all peripherals
void init_devices(void)
{
 //stop errant interrupts until set up
 CLI(); //disable all interrupts
 port_init();
 MCUCR = 0x00;
 TIMSK = 0x00; //timer interrupt sources
 GIMSK = 0x00; //interrupt sources
 SEI(); //re-enable interrupts
 //all peripherals are now initialized
}
void writeE(unsigned char adr, unsigned char data)
{
  while (EECR & BIT(EEWE)); // Wait until EEWE flag is cleared
  EEAR = adr;                  // Load address
  EEDR = data;                 // Load data
  EECR = BIT(EEMWE);           // EEPROM write enable
  EECR |= BIT(EEWE);
}
```

```
//Основна функціія:
void main(void)
{
  unsigned char a,i,j=0;
  while(1)
  {
    for(i=j; i<=64+j; i++)
    {
                a = RAND_LOOKUP[i];
                if (a < 127) asm("nop\n");
                PORTB = a;
    }
    writeE(j, a);
    j++;
  }
}
```

Скомпільований HEX-файл займає 428 байт, що становить 42% загального обсягу пам’яті. Враховуючи 256 байт статичних даних сама програма становить лише 172 байти циклічного коду, що відповідає критерію мінімального використання ресурсів АЛП. Виконання зсуву адреси записуваної комірки EEPROM та її одиничний перезапис впродовж тривалого циклу забезпечують досягнення максимальної швидкодії і довготривалого використання EEPROM, що має досить обмежений ресурс перезапису.

Виконання останніх двох програм мовою програмування високого рівня робить її зрозумілими та за необхідності легко модифікованими користувачем навіть без участі розробника.

5.4. Структурна схема експериментальної установки і результати експерименту

Експериментальна установка (рис.5.6) призначена для виконання програмування мікроконтролерів тестового макету, отримання та переведення в цифровий вигляд даних струмів споживання мікроконтролера із захистом та

завантаження цих даних у комп’ютер з метою подальшої обробки з використанням програмного забезпечення, розробленого в розділі 4.

Для зручності програмування в якості програматора використаний STK-500 AVR/ISP, що має USB-інтерфейс і є одним зі стандартних пристроїв для програмування як в AVR Studio, так і в різних версіях ImageCraft, що використовувалися при написанні програмного забезпечення для тестового макету.

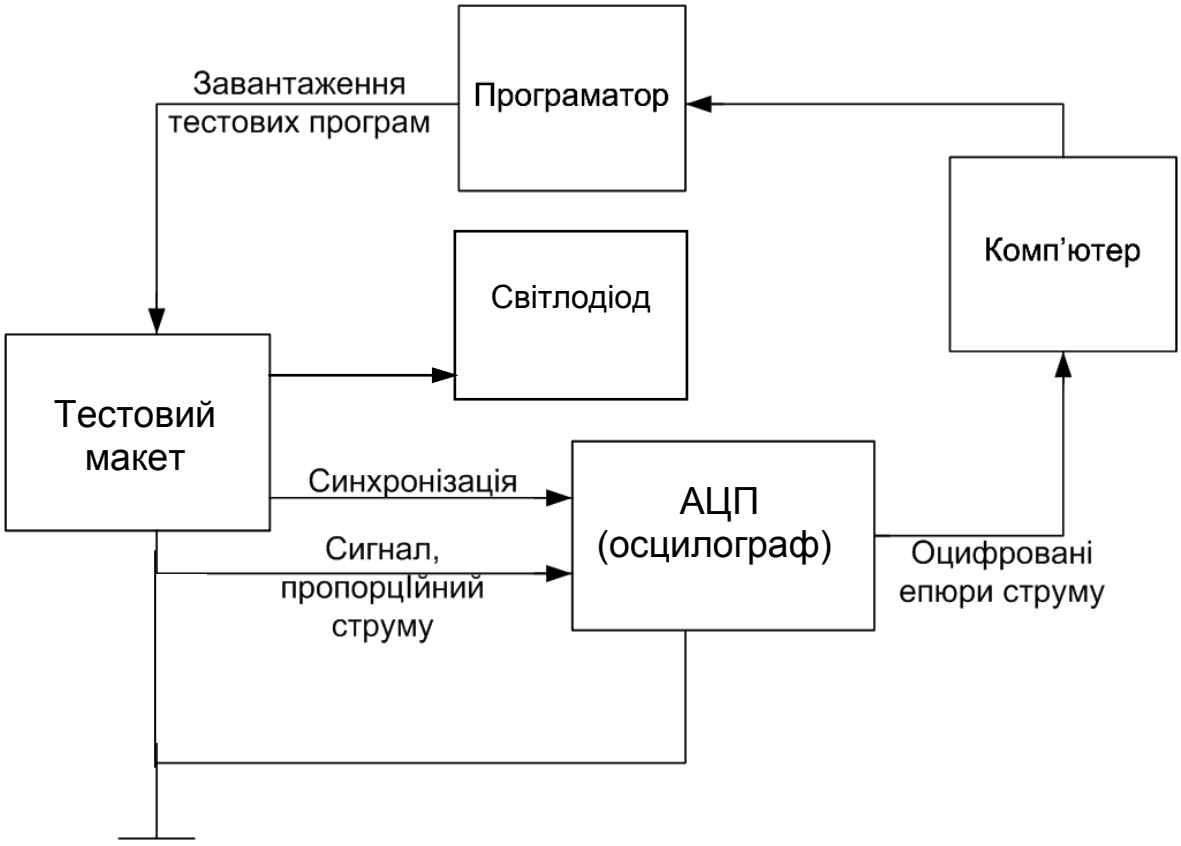


Рис 5.6. Структурна схема експериментальної установки.

З датчика струму сигнал знімається за допомогою каналу «В» цифрового осцилографа. Канал «А» осцилографа використовується для підключення сигналу синхронізації, що також генерується мікроконтролером. Сигнал синхронізації введено для отримання точки відліку у сигналі струму споживання.

Для отримання коректних результатів, база знятих даних струмів споживання повинна мати наступні складові:

1) наявність масиву 1000 зразків струмів споживання для кожної підпрограми;

2) усереднені струми споживання, що відповідають усуненню завад;

3) осцилограми завад генерованих системами захисту;

4) по 100 зразків струмів споживання мікроконтролера з кожною системою захисту для кожної підпрограми.

Наявність осцилограм в базі даних дає можливість візуально оцінити як різницю в струмах споживання мікрокоманд, так і ефективність кожної перевіреної системи захисту.

Можливість розрізнення струмів споживання мікрокоманд та підпрограм підтверджує суміщений графік струмів споживання для семи досліджуваних алгоритмів (рис.5.7).

На рис.5.8. на меншому масштабі часу показано область «А» з вищенаведеного рис 5.7. Візуальний аналіз показує, що при виконанні різних інструкцій МП споживає динамічний струм різної форми, причому можна виділити не лише різницю в амплітудному значенні стрибків та провалів, а й помітити різницю в інтенсивності переключення ключів, що відображається як періодичні стрибки протягом одного машинного такту.

Використані підпрограми розроблялися з метою отримання як схожих, так і суттєво різних струмів споживання, однак з рис.5.7 слідує, що даний аналіз придатний навіть для розпізнання мікрокоманд на рівні операндів. Співпадіння знятих осцилограм за імпульсами синхронізації свідчить про високий рівень можливостей експериментальної установки.

Згідно графіків (рис.5.7-5.8) доцільним є проведення детального аналізу корельованості знятих струмів споживання для отримати числових значень характеристик.

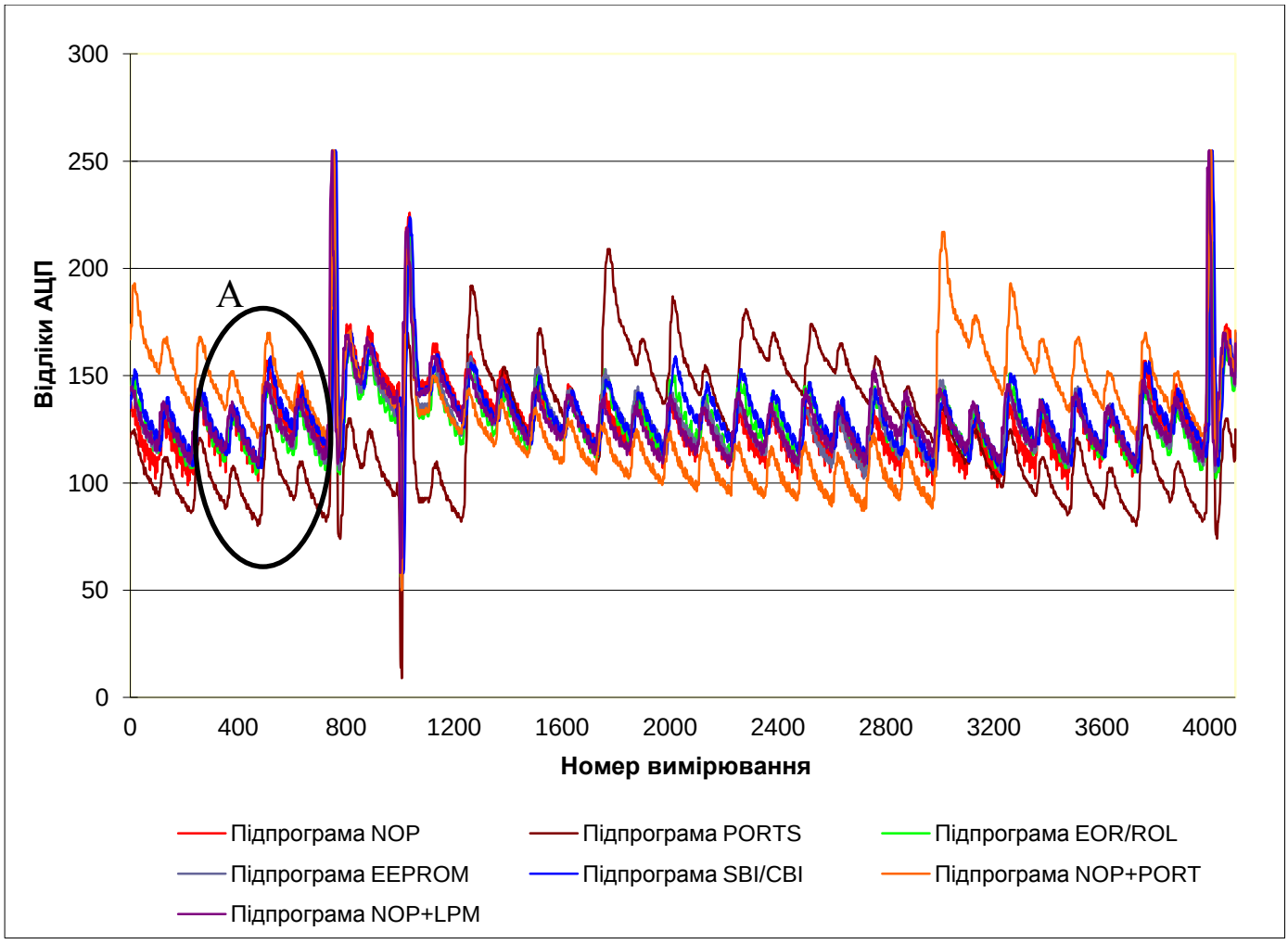


Рис.5.7. Графічне зображення струмів споживання семи підпрограм основного мікроконтролера.

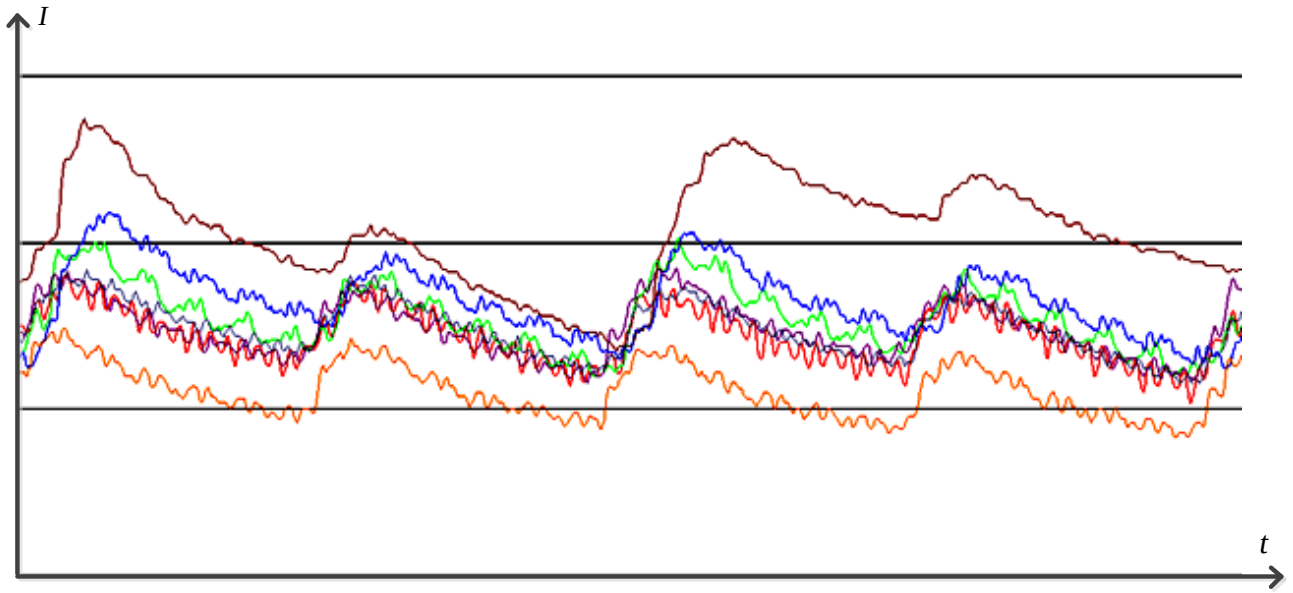


Рис.5.8. Збільшене зображення струмів споживання при виконанні підпрограм.

П’ять з семи досліджуваних систем захисту генерують шум за струмом споживання. Для кожної з них отримана своя осцилограма, що характеризує роботу системи та її теоретичну ефективність.

На рис.5.9. зображена осцилограма шуму регульованого фільтра зі змінними параметрами.

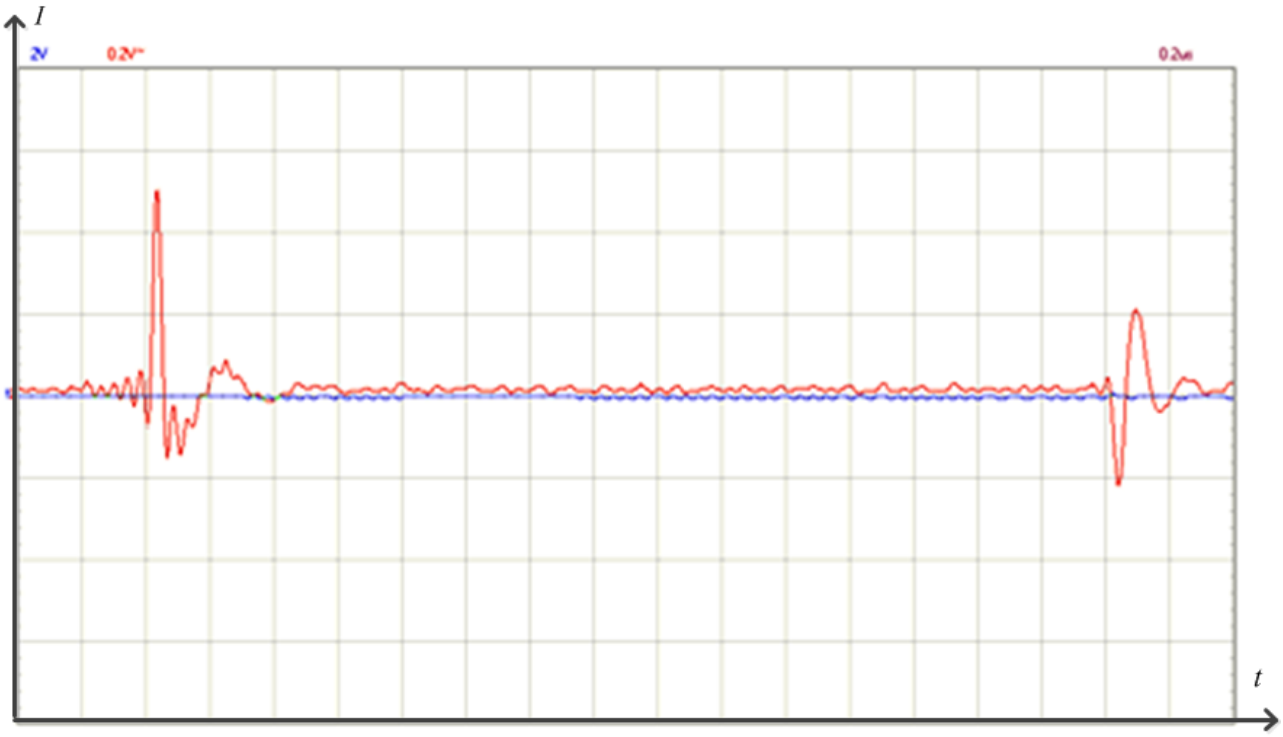


Рис.5.9. Осцилограма впливу на струм споживання фільтру
зі змінними параметрами.

Чітко видно два перехідні процеси підключення і відключення конденсатора різної тривалості. Причина цього полягає в зміні опору фільтру, що на певному періоді представляє собою LC-фільтр, а на іншому L-фільтр. Відповідно постійні часу не рівні між собою. Обидва перехідні процеси коливальні, тому їхній вплив на динамічний струм повинен бути помітнішим, ніж при умові аперіодичних перехідних процесів.

Для **Регульованого фільтру 2** на основі блоку ключів, що змодельовані резисторами отримано осцилограму рис.5.10. Вона характеризується більшим

періодом та меншою величиною пульсацій. Дуже незначний їх рівень пояснюється незавершеністю перехідного процесу розряду ємностей затворів транзисторів, що працюють на близькій до граничної частоті та не мають спеціальної системи розряду. Їхній розряд здійснюється лише через порт мікроконтролера. Збільшення імпульсів струму вимагає зменшення опорів резисторів до такого рівня, коли їхній струм буде співвимірний зі струмом споживання мікроконтролера, що захищається. При цьому на них та внутрішніх опорах відкритих ключів буде розсіюватись теплота еквівалентна всій потужності мікроконтролера, що вимагає виконання ключів та їх навантажень за іншою більш потужною технологією.

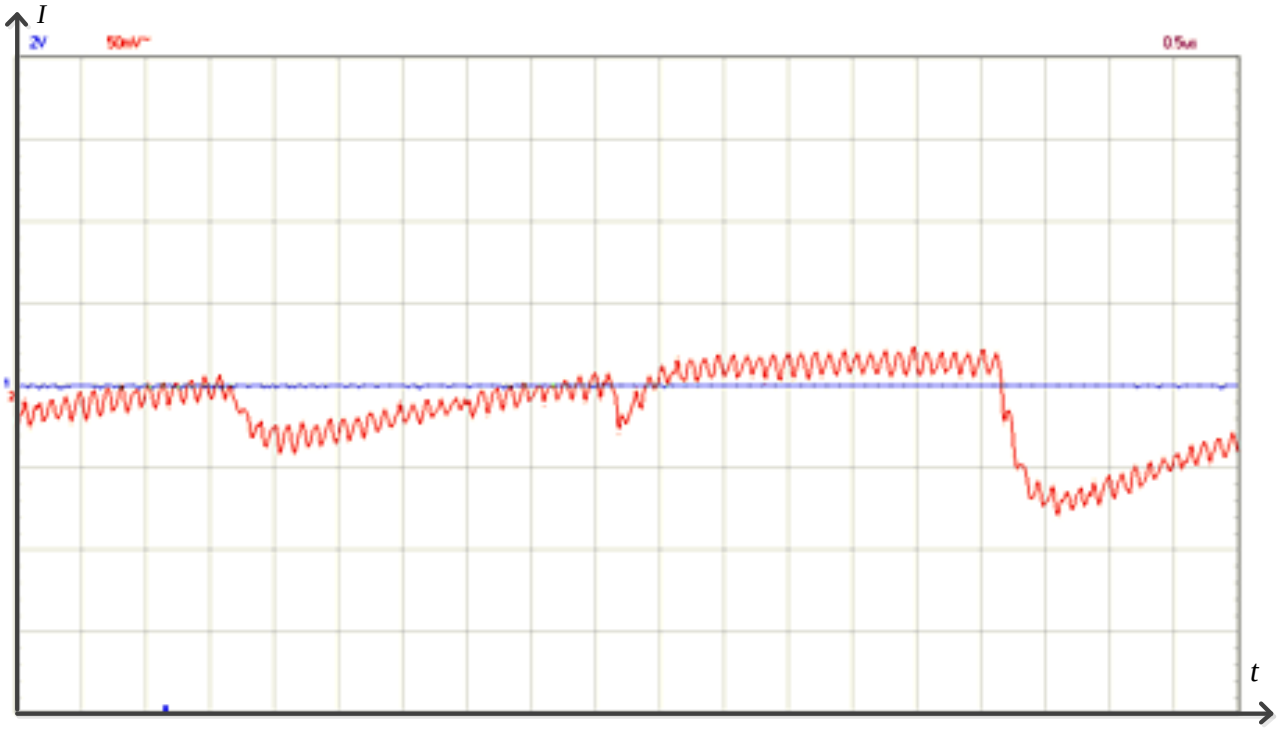


Рис.5.10. Осцилограма впливу на струм споживання комутованих резисторів.

Проведена модифікація тестового макету полягає в підключенні до двох виводів портів мікроконтролера системи керування постійного конденсатора ємністю 20 пФ (рис 5.11), модифікації програмного забезпечення

мікроконтролера таким чином, щоб на портах сигнали змінювалися у протифазі – це спричиняє перезарядку конденсатора від джерела живлення, та значні сплески у струмі споживання.

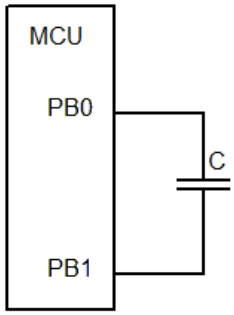


Рис 5.11. Схема підключення постійного конденсатора до портів мікроконтролера

Отримана осцилограма струму споживання тестового макету після модифікації зображена на рис. 5.12. Наведена осцилограма ілюструє накладання аперіодичного процесу заряду конденсатора на струм споживання системи керування його перезарядкою. Діаграми свідчать, що усунення шляхом цифрової обробки зумовленого конденсатором перехідного процесу дасть струм споживання близький до вихідного.

Захист на основі маніпуляції внутрішніми ресурсами забезпечує додавання хаотичної складової струму з тією ж самою частотою та амплітудою (рис.5.13), однак на практиці цей метод потребує розподілення ресурсів МП на періодичну ініціалізацію внутрішніх ресурсів, що можуть в цей час бути вже задіяними, що обмежує практичне використання даної системи захисту.

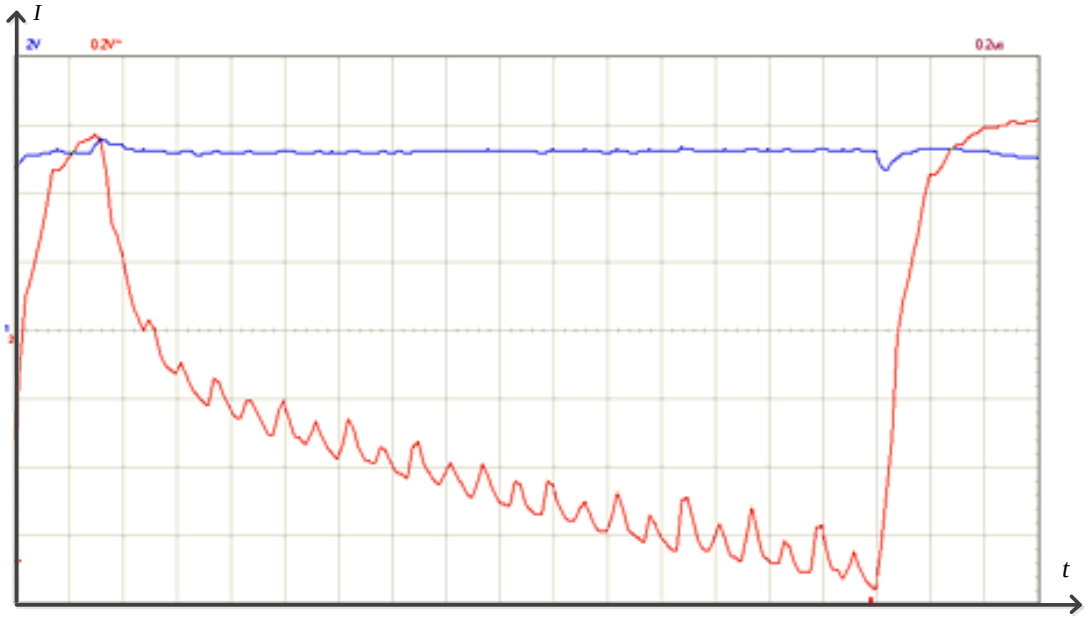


Рис.5.12. Осцилограма струму споживання мікроконтролера, що виконує примусову перезарядку конденсатора.

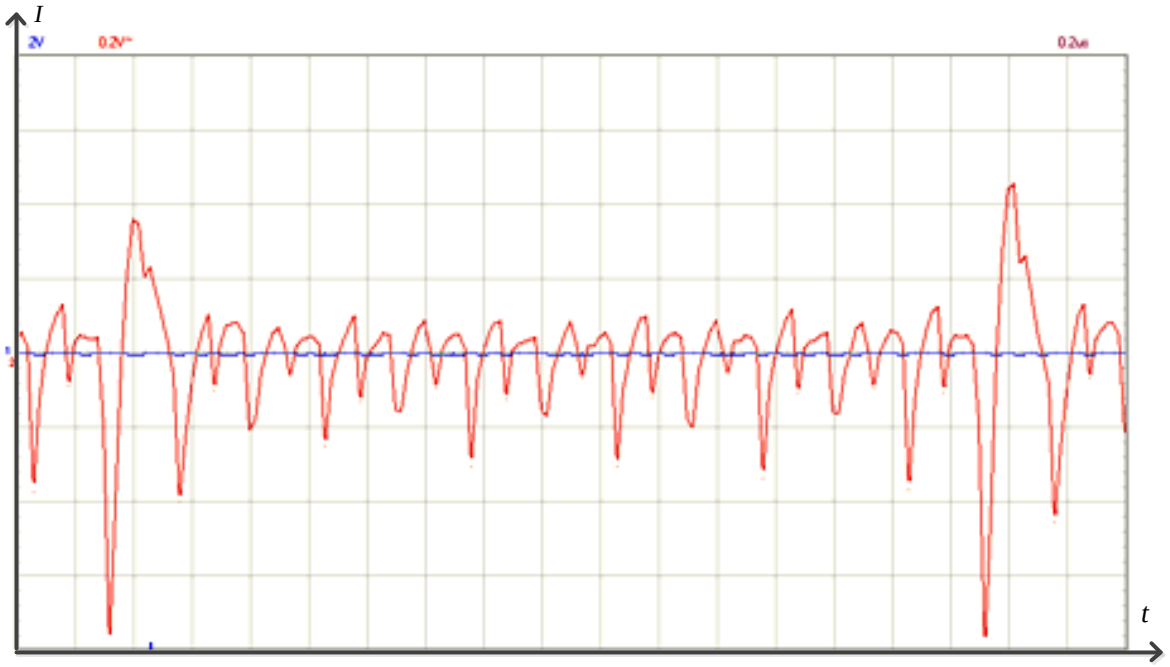


Рис.5.13. Осцилограма струму споживання мікроконтролера при включенні-виключенні інтегрованих пристроїв.

Використання додаткового мікроконтролера, що моделює допоміжне ядро створює завади за струмом споживання представлені на рис.5.14. Частота генерованих ним шумів є максимальною серед розглянутих систем.

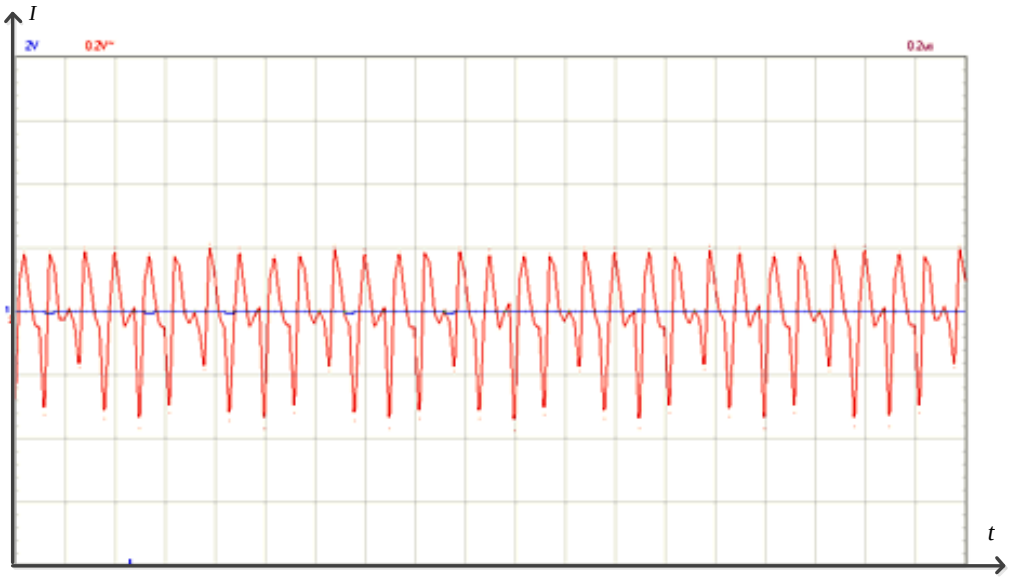


Рис.5.14. Осцилограма струму споживання захисного мікроконтролера.

Накладання останніх двох струмів споживання повинно суттєво впливати на результуючий струм, що потребує подальшого детального аналізу з використанням математичного апарату.

Зняття бази даних проводилося відповідно до раніше розробленої принципової схеми з однаковою тактовою частотою роботи мікроконтролерів. Оскільки більшість мікропроцесорних систем мають лише одну тактову частоту, що зазвичай є максимально можливою, то вплив кратності та ірраціонального співвідношення частот потребує додаткових досліджень.

5.5. Розробка програмного забезпечення для кореляційного аналізу

Як було відмічено раніш (див. п. 3.5)Визначення ступеня ефективності різних систем захисту від атак за струмом споживання проводиться шляхом їх порівняльного аналізу.

Оскільки оцінка та порівняння між собою великого масиву даних в ручному режимі вимагає багато часу, для спрощення цієї задачі написано програмне забезпечення в системі MatLab. Програма дозволяє обробляти 1000 вихідних струмів споживання при виконанні кожної підпрограми без системи захисту та проводить їхнє усереднення. Дослідження всіх систем захисту проводиться з використанням 100 кривих струмів споживання при виконанні підпрограми, усереднення при цьому не проводиться.

Використання системи MatLab дозволяє значно спростити вихідний текст програми за наявності великої кількості вбудованих функцій для роботи з векторами, матрицями та математичними виразами. Ще однією перевагою MatLab є досить проста візуалізація даних у вигляді двовимірних та тривимірних графіків, поверхонь та ін. Недоліком системи MatLab можна назвати невисоку швидкість обчислень. Обробка великих масивів даних може займати багато часу навіть на сучасних комп'ютерах. Саме цей недолік і перешкоджає використанню MatLab у системах обробки даних реального часу.

Складовими частинами розробленого програмного забезпечення є три програми, що на виході дають таблиці спряженості. Перша програма переписує з бази даних дискретизовані значення струмів споживання у формат необхідний для подальшої обробки – формує вектори синхронізації і числових значень струмів. Виділення в кожному записаному файлі чітких меж струму споживання за імпульсом синхронізації призначена друга програма. Критерієм, що визначає границю імпульсу синхронізації є відхилення його амплітудного значення на 10 одиниць або може бути змінене на будь-яке інше при проведенні дослідження на іншому масштабі вимірювання. Таким чином відбувається виділення синхронізованих з точністю до кроку дискретизації (2мкс) струми споживання при виконанні підпрограм.

Основна програма виконує всі необхідні обчислення коефіцієнтів кореляції та зводить отримані результати до зручного табличного вигляду. П'ятнадцять вихідних таблиць відповідають кореляції струмів споживання без систем

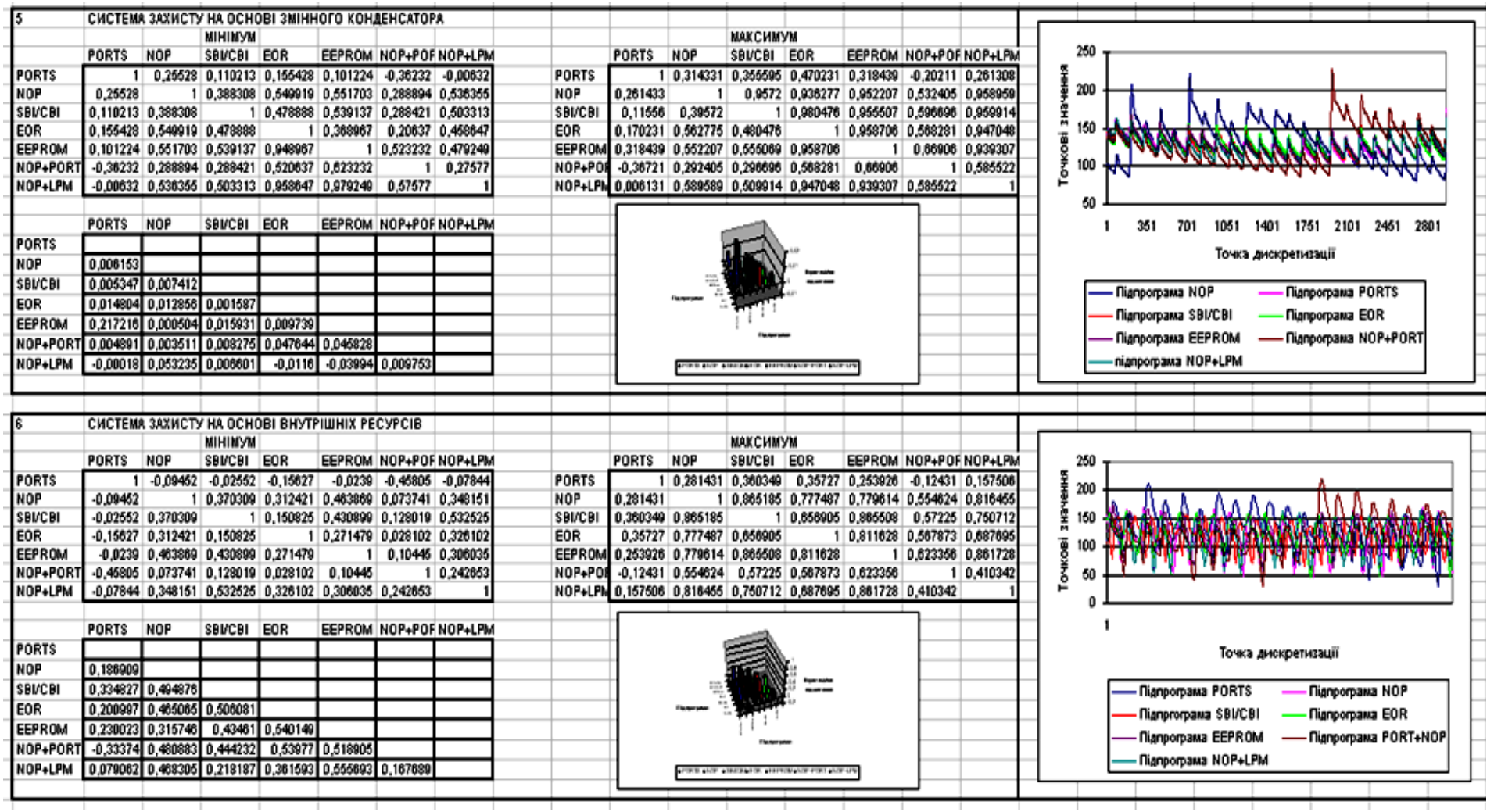
захисту та мінімальному і максимальному значенням коефіцієнтів кореляції для кожної з семи систем захисту.

Внаслідок тривалого часу обробки програмами Matlab бази даних, подальший аналіз вирішено проводити в середовищі Excel, що оптимізоване для роботи з табличними даними. Зовнішній вигляд вікна програми збору, візуалізації та оцінки отриманих результатів наведено на рис.5.15. Для кожної системи захисту будується три таблиці, що відповідають:

1) мінімальним значенням коефіцієнтів взаємної кореляції при дослідженні 100 зразків струмів споживання взятих в довільні моменти часу;

2) мінімальним значенням відповідних коефіцієнтів визначених аналогічно до першої таблиці;

3) розраховані відхилення коефіцієнтів кореляції.



Таблиці значень Тривимірна діаграма результатів Суміщені графіки

Рис.5.15. Програма кінцевої обробки результатів в середовищі Excel

5.6. Порівняльний аналіз ефективності використання регульованих фільтрів

Аналіз даних розпочнемо з вихідних даних струмів споживання мікроконтролера без систем захисту. Отриманим суміщеним графікам (рис.5.3) відповідає таблиця коефіцієнтів кореляції табл.5.2.

Таблиця. 5.2

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS	1						
NOP	0,118593	1					
SBI/CBI	0,346476	0,660084	1				
EOR	0,450144	0,815312	0,792209	1			
EEPROM	0,294136	0,88862	0,733365	0,935724	1		
NOP+PORT	-0,47184	0,277739	0,173997	0,281835	0,393816	1	
NOP+LPM	0,118147	0,892588	0,561757	0,847612	0,859811	0,376784	1

Відповідно до табл.5.2. побудована тривимірна гістограма розподілу коефіцієнтів кореляції вихідних струмів споживання (рис.5.16).

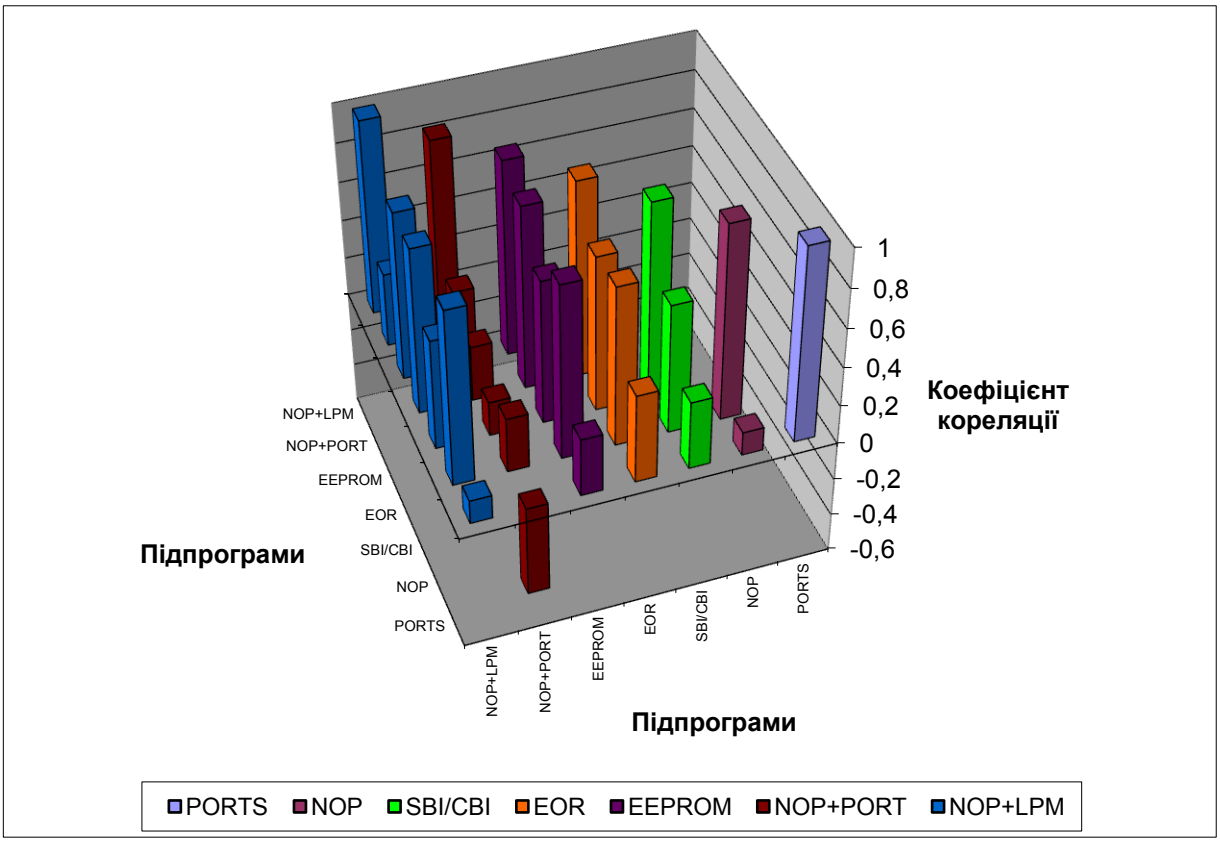


Рис.5.16. Гістограма коефіцієнтів кореляції вихідних струмів.

З рис.5.2 видно, що підтверджується гіпотеза, зроблена на основі візуального спостереження осцилограм при їх вимірюванні – виконувані підпрограми корелюють між собою з коефіцієнтом кореляції меншим за одиницю, а, отже, їх можна розрізнити з проведенням такого дослідження в часовій області. Діапазон зміни коефіцієнтів кореляції від -0,47 до 0,93 свідчить, що підпрограми розпізнаються з різним ступенем складності – чим менше отримане значення та висота стовпчика гістограми в сторону від'ємних значень, тим ефективніше здійснюється розрізнення. Як і у випадку суміщеного графіку (рис.4.3) найкраще розпізнаються підпрограми PORT та NOP+PORT.

Подальший аналіз включає почергове дослідження модельованих систем захисту з метою визначення їх ефективності.

Детально розглянемо оцінку ефективності першого досліджуваного регульованого фільтру на основі фільтруючого конденсатора. Для неї отримані суміщені графіки (рис.5.17) та числові значення мінімальних (табл. 5.3) і максимальних (табл. 5.4) коефіцієнтів кореляції.

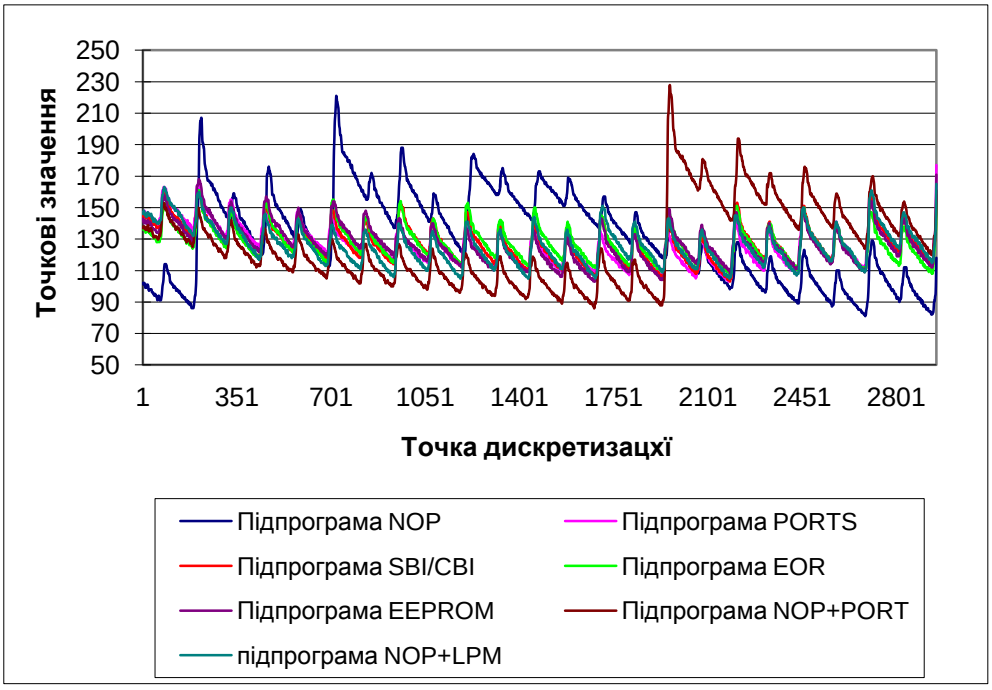


Рис.5.17. Суміщені графіки для аналізу системи захисту на основі вхідного фільтруючого конденсатора.

Таблиця 5.3

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS	1						
NOP	0,104083	1					
SBI/CBI	0,164325	0,921541	1				
EOR	0,411262	0,817842	0,922448	1			
EEPROM	0,185096	0,925737	0,921073	0,873579	1		
NOP+PORT	-0,49831	0,251503	0,340123	0,27247	0,40889	1	
NOP+LPM	0,007569	0,888507	0,91571	0,823354	0,854085	0,386413	1

Таблиця 5.4

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS	1						
NOP	0,120048	1					
SBI/CBI	0,171682	0,930386	1				
EOR	0,418088	0,830464	0,925754	1			
EEPROM	0,19424	0,930818	0,923227	0,879905	1		
NOP+PORT	-0,49403	0,257524	0,344279	0,280346	0,417645	1	
NOP+LPM	0,01638	0,898454	0,917761	0,82925	0,857852	0,392856	1

Для точної математичної оцінки розрахуємо відхилення коефіцієнтів кореляції, що створюються даною системою захисту. Для цього використаємо формулу:

$$\Delta_{ij} = p_{ij\max} - p_{ij\min} \qquad (5.1)$$

де p_{ij} – відповідний коефіцієнт кореляції розташований в і-тому рядку і j-тому стовпчику.

Таблиця відхилень для розглянутої системи (табл..5.5) будується з урахуванням симетричності та видалення нульових елементів.

Таблиця 5.5

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,015965					
SBI/CBI	0,007357	0,008844				
EOR	0,006826	0,012621	0,003306			
EEPROM	0,009144	0,005081	0,002154	0,006326		
NOP+PORT	-0,00429	0,006021	0,004156	0,007877	0,008754	
NOP+LPM	0,008811	0,009947	0,002052	0,005896	0,003768	0,006443

Відповідно до даних табл..5.5 будуємо тривимірну гістограму (рис.5.18), що відображає ефективність впливу на початкові коефіцієнти кореляції.

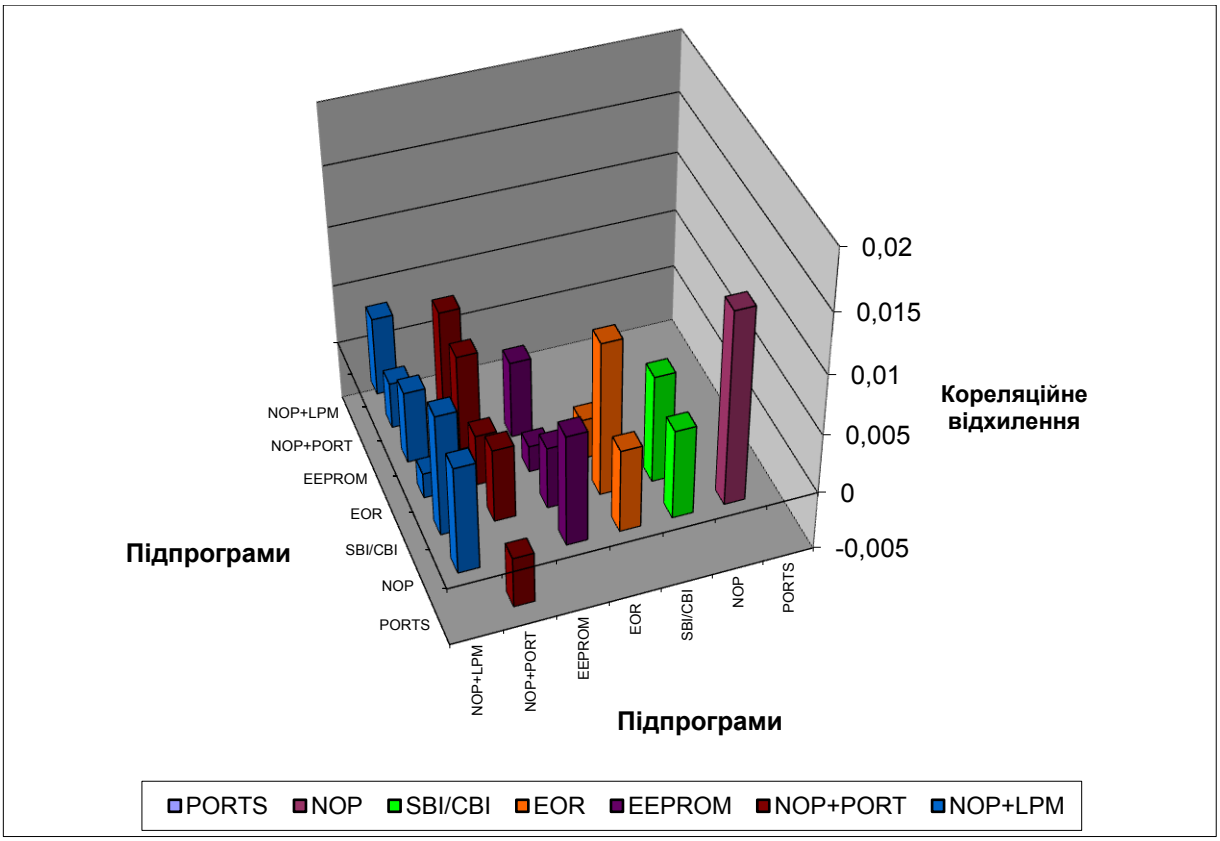


Рис.5.18. Гістограма відхилень системи захисту
на основі вхідного фільтруючого конденсатора.

Визначимо максимальне отримане відхилення використовуючи формулу, що випливає з (5.1):

$$\Delta_{\max} = \max(p_{ij\max}) - \min(p_{ij\min}) \qquad (5.2)$$

Обчислений для розглянутої системи захисту коефіцієнт $\Delta_{\max} = 0{,}02$, що відповідає 2% ефективності впливу щодо зменшення здатності отримання розпізнання.

Надалі будемо аналізувати системи захисту за спрощеною методикою наводячи лише суміщені графіки струмів споживання при виконанні підпрограм, таблицю відхилень коефіцієнтів кореляції з побудованою відповідно до неї гістограмою та числові значення оцінки ефективності.

Враховуючи наявність періодичних імпульсів, що формує *система захисту на основі фільтру зі змінними параметрами*, вона повинна бути більш

ефективною, ніж попередньо розглянута. Для неї маємо графіки (рис.5.19) та дані з табл..5.6.

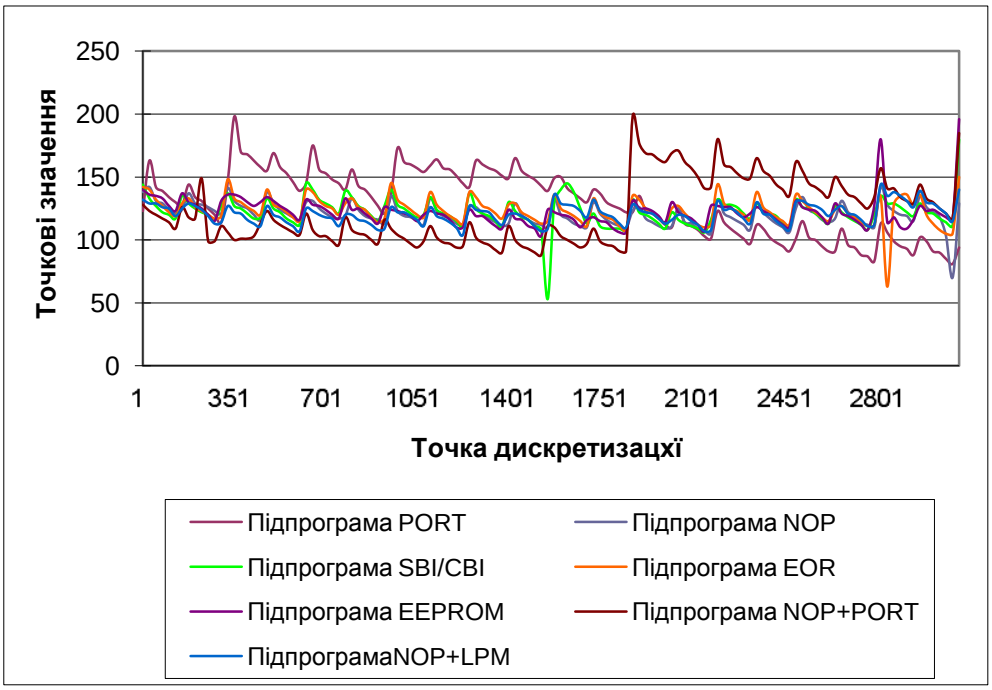


Рис.5.19. Суміщені графіки для аналізу системи захисту на основі фільтру зі змінними параметрами .

Таблиця 5.6

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,221814					
SBI/CBI	0,194213	0,147098				
EOR	0,163747	0,277891	0,308494			
EEPROM	0,105009	0,225019	0,214363	0,159483		
NOP+PORT	-0,06183	0,095214	0,161641	0,189355	0,15168	
NOP+LPM	-0,04025	0,223629	0,280195	0,257238	0,259034	0,109146

Отримані числові значення візуалізуємо на рис.5.20.

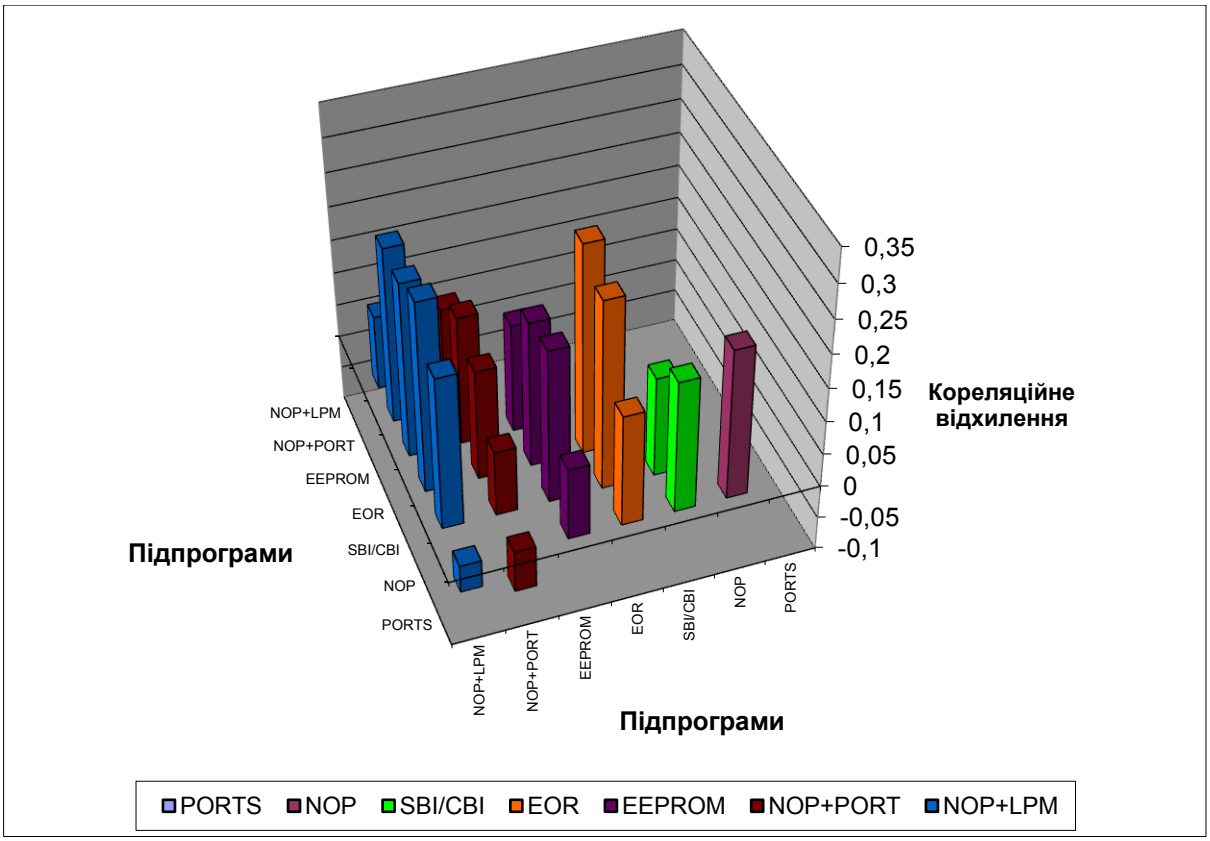


Рис.5.20. Гістограма відхилень системи захисту
на основі фільтру зі змінними параметрами.

Розраховуємо максимальне відхилення коефіцієнтів кореляції рівне $\Delta_{max} =$ 0,37. Тобто дана система дозволяє зменшити ймовірність детектування струмів споживання, що виникають при виконанні мікрокоманд та підпрограм мікроконтролера на 37%.

Наступній системі захисту зі стабілізатором напруги відповідають рис.5.21 та табл.5.7.

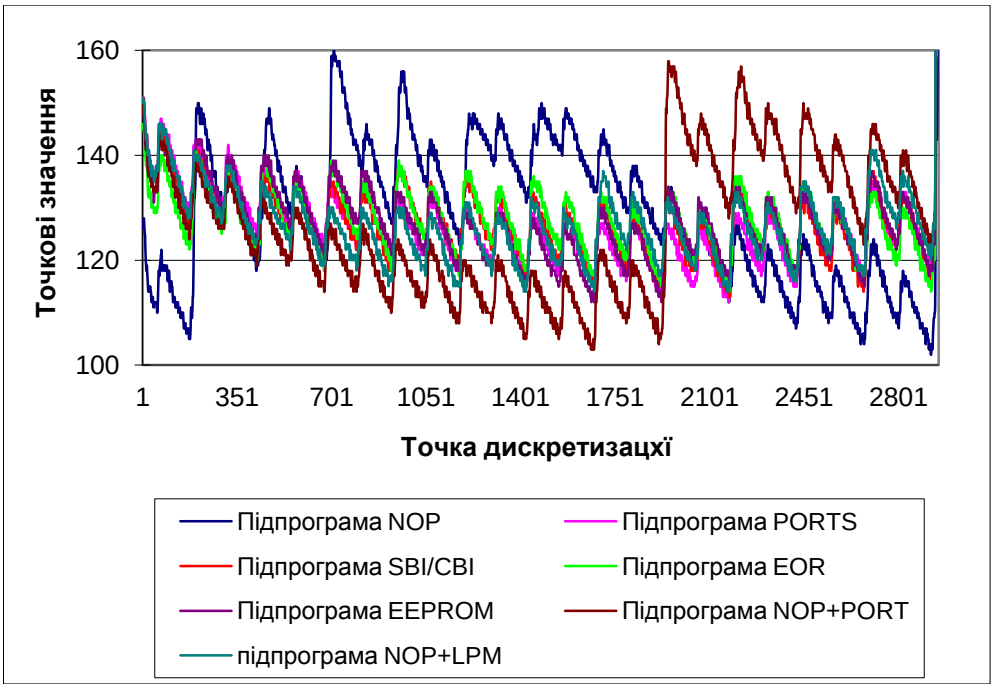


Рис.5.21. Суміщені графіки для аналізу системи захисту зі стабілізатором напруги.

Таблиця 5.7

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,010889					
SBI/CBI	0,012769	0,005561				
EOR	0,010789	0,007702	0,004503			
EEPROM	0,014653	0,004307	0,002113	0,007802		
NOP+PORT	-0,00951	0,009856	0,015505	0,015821	0,009965	
NOP+LPM	0,018319	0,012836	0,008258	0,010027	0,00781	0,012995

Відповідно до представленої гістограми (рис.5.22) можна зробити висновок про найбільшу хоч і незначну ефективність методу стабілізації напруги для маскування підпрограм, що містять спільну складову (таких як алгоритми шифрування), оскільки чітко спостерігаються максимуми для підпрограм, що містять мікрокоманду NOP.

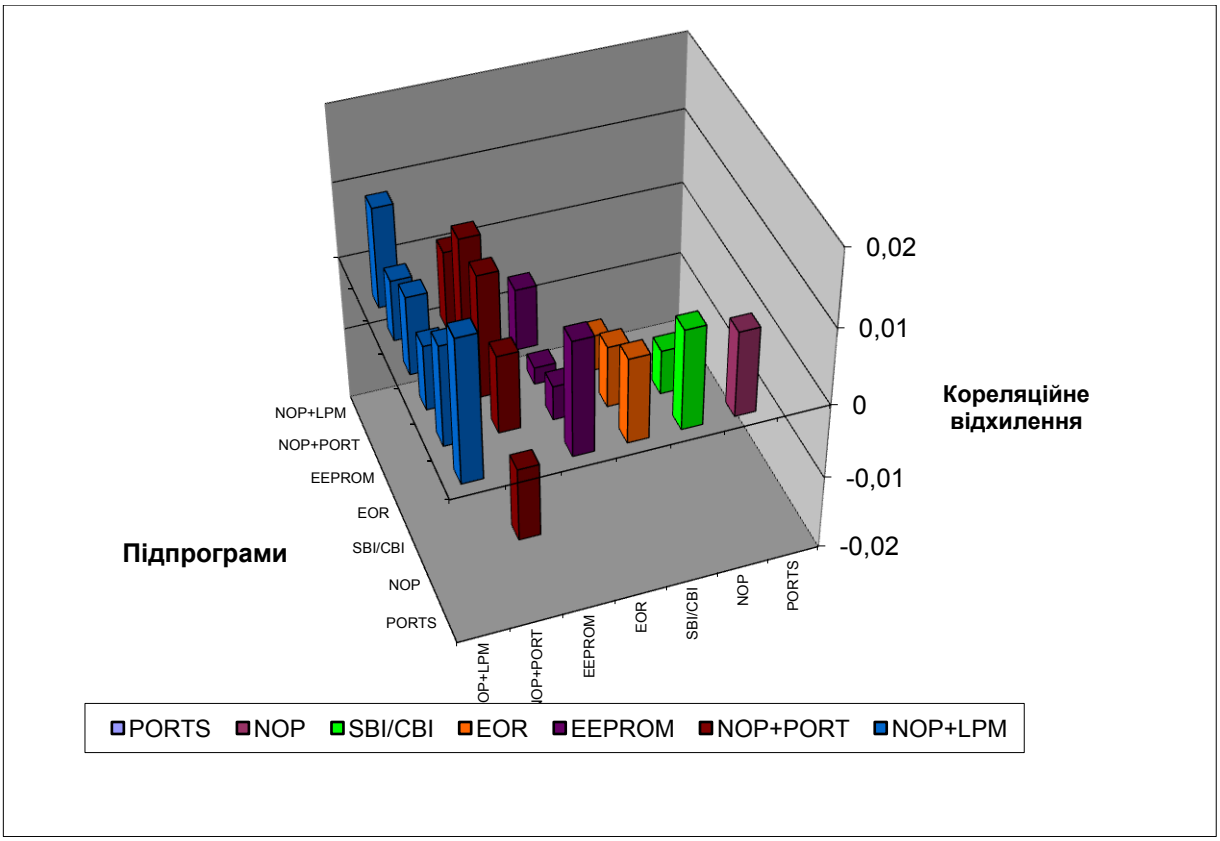


Рис.5.22. Гістограма відхилень системи захисту
зі стабілізатором напруги.

Розраховуємо максимальне відхилення коефіцієнтів кореляції рівне Δ_{max} = 0,028, якому відповідає зменшення ймовірність детектування струмів споживання на 2,8%, що є кращим ніж у першої системи захисту від атак за струмом споживання, однак не достатнє для проведення подальших розробок в цьому напрямку.

Змодельовані *системи захисту на основі блоку ключів* та *змінного конденсатораі* мають схожі характеристики. Їхні графіки суміщених струмів споживання слабко відрізняються від наведених на рис5.3, тому немає сенсу їх повторного приведення.

Таблиці відхилень коефіцієнтів (табл.5.8-5.9) та гістограми (рис.5.23-5.24) відповідають системі захисту на основі блоку ключів та змінного конденсатора відповідно.

Таблиця 5.8

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,010735					
SBI/CBI	0,005412	0,5795				
EOR	-0,02903	-0,00224	0,005704			
EEPROM	0,080157	0,000527	-0,00171	-0,02862		
NOP+PORT	-0,15451	0,045989	-0,03332	-0,01053	0,00232	
NOP+LPM	0,087326	0,194113	-0,00182	0,000306	-0,02569	0,002663

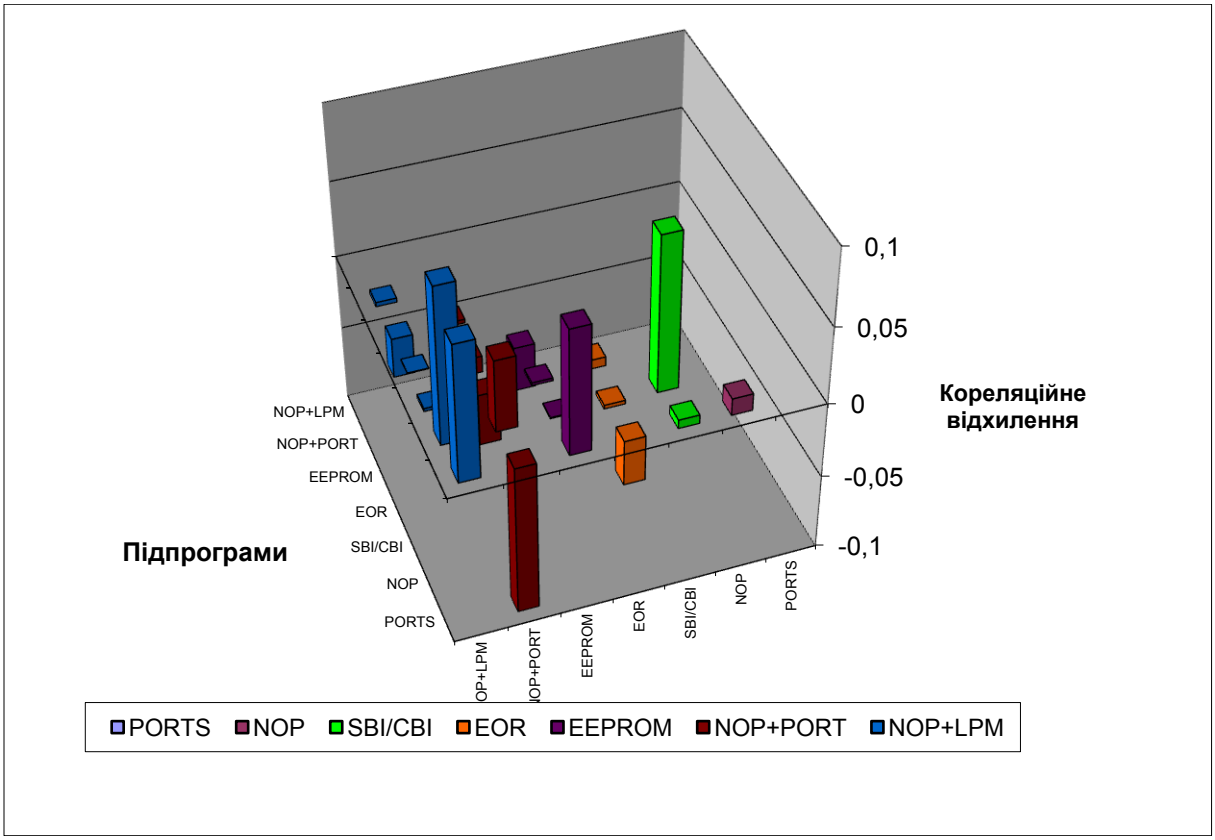


Рис.5.23. Гістограма відхилень системи захисту на основі блоку ключів.

Таблиця 5.9

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,010889					
SBI/CBI	0,012769	0,005561				
EOR	0,010789	0,007702	0,004503			
EEPROM	0,014653	0,004307	0,002113	0,007802		
NOP+PORT	-0,00951	0,009856	0,015505	0,015821	0,009965	
NOP+LPM	0,018319	0,012836	0,008258	0,010027	0,00781	0,012995

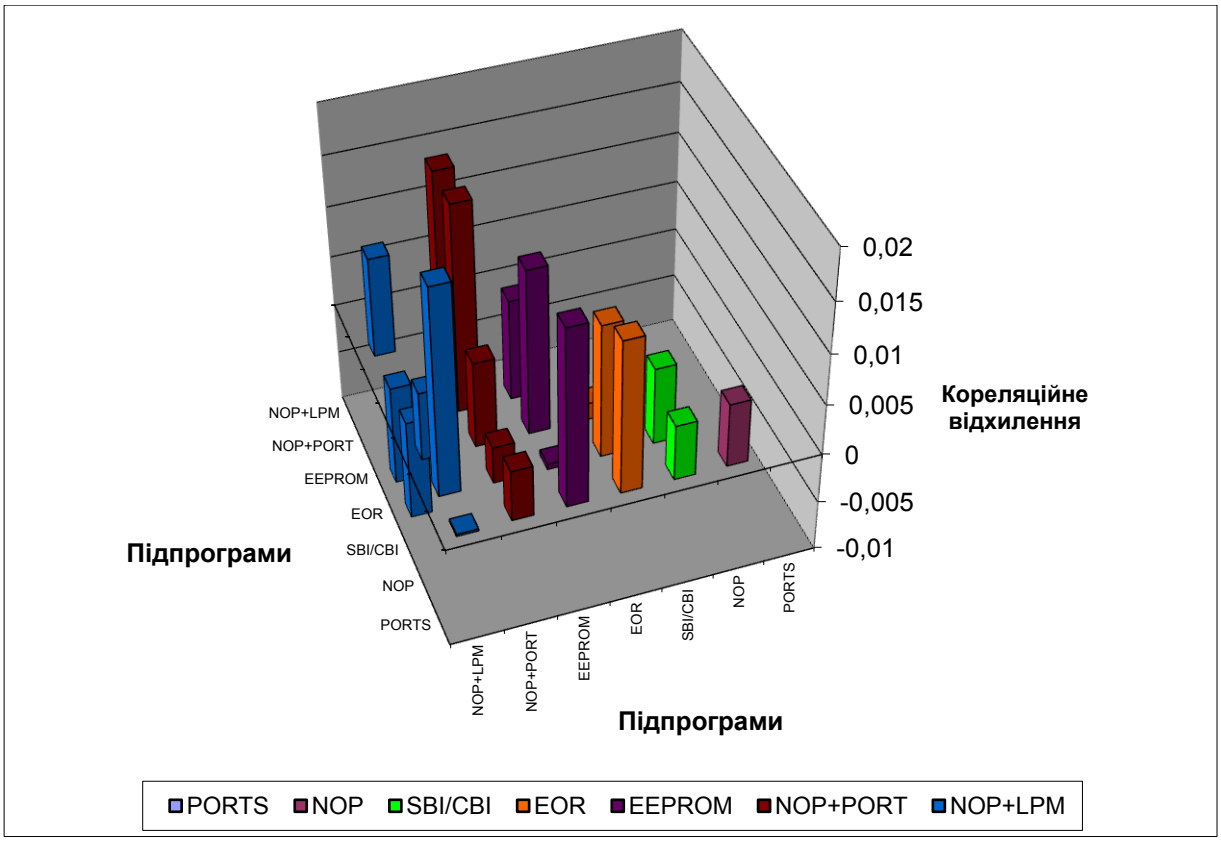


Рис.5.24. Гістограма відхилень системи захисту
на основі змінного конденсатора.

Незначні точкові відхилення, що спостерігаються на гістограмах свідчать про вірогідність впливу завад на процес вимірювання, оскільки виділення якої-небудь закономірності в роботі систем захисту, що розглядаються, неможливо. Для збереження послідовності аналізу розрахуємо максимальні відхилення Δ_{max} для цих систем. Вони рівні 0,12 та 0,09, тобто відповідають 12% та 9% ефективності.

Візуальний аналіз, проведений для систем захисту, що базуються на основі допоміжного ядра і маніпуляції внутрішніми ресурсами, і мають у своєму складі генератори випадкових чисел, показав необхідність визначення ефективнішої з них. Додатково наведемо для них суміщені графіки трьох струмів споживання при виконанні однієї підпрограми (рис 5.25-5.26). Для прикладу візьмемо підпрограму PORT+NOP.

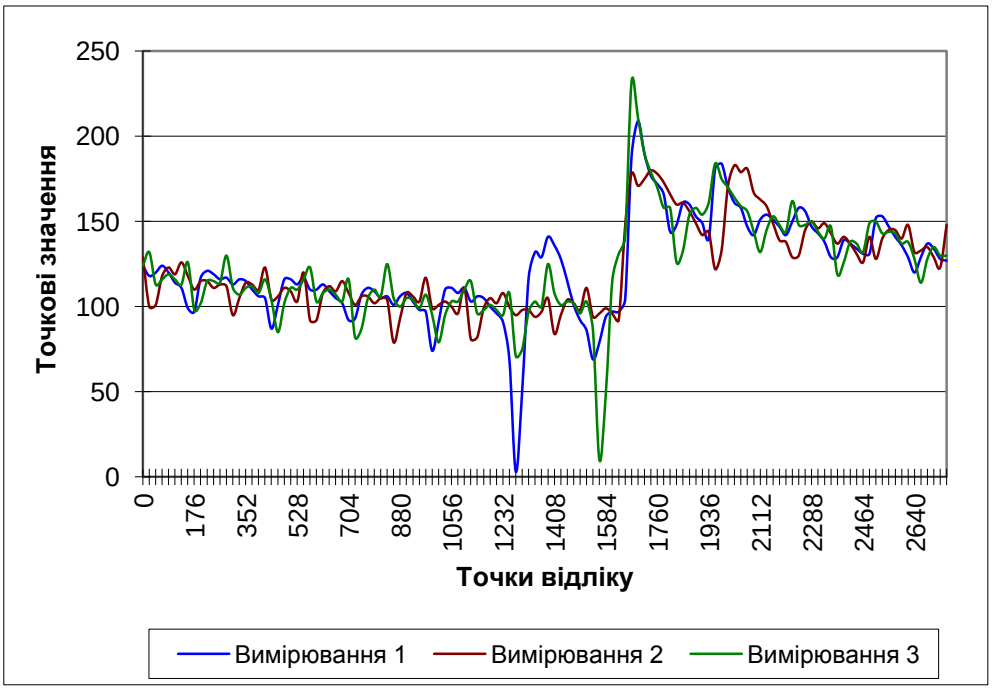


Рис.5.25. Суміщені графіки виконання однієї підпрограми мікроконтролера з системою захисту на основі маніпуляції внутрішніми ресурсами.

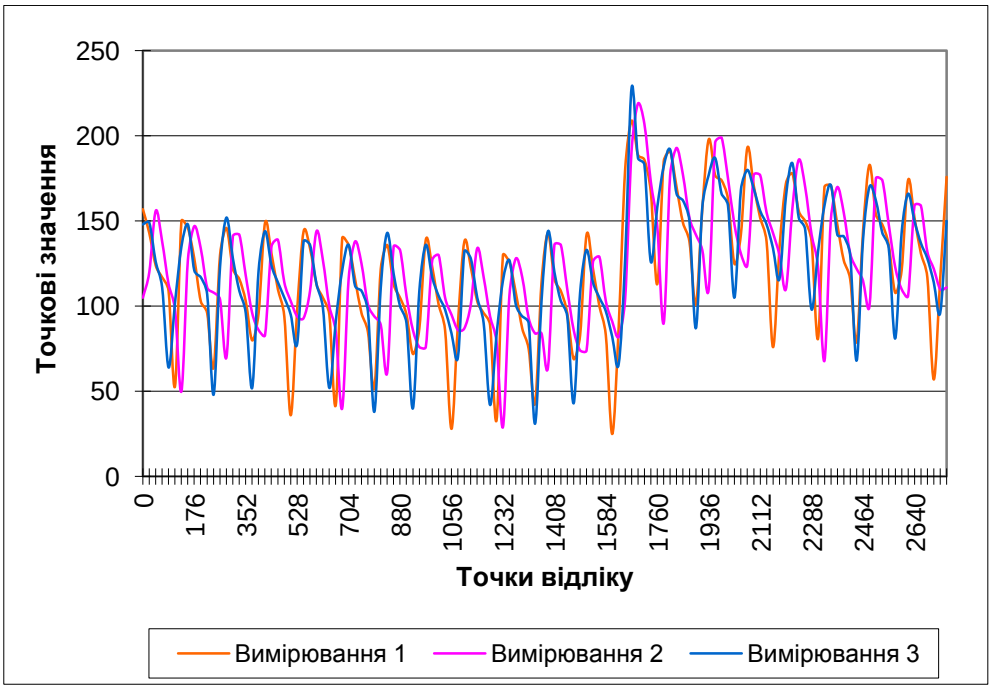


Рис.5.26. Суміщені графіки виконання однієї підпрограми мікроконтролера з системою захисту на основі допоміжного ядра.

Отриманий графік (рис.5.24) показує значне спотворення фази сигналу і незначну зміну амплітудних значень викидів для системи захисту на основі маніпуляції внутрішніми ресурсами, причому простежується вихідна форма знятого струму споживання без системи захисту. Графік на рис.5.25 свідчить про рівномірну фазову та імпульсну деформацію струму споживання та приведення його до більш рівномірного вигляду.

Проведемо подальше дослідження згідно запланованої послідовності. Для цього приведемо суміщені графіки (рис.5.27) та таблицю відхилення (табл.5.10) для *системи захисту на основі маніпуляції внутрішніми ресурсами* мікроконтролера.

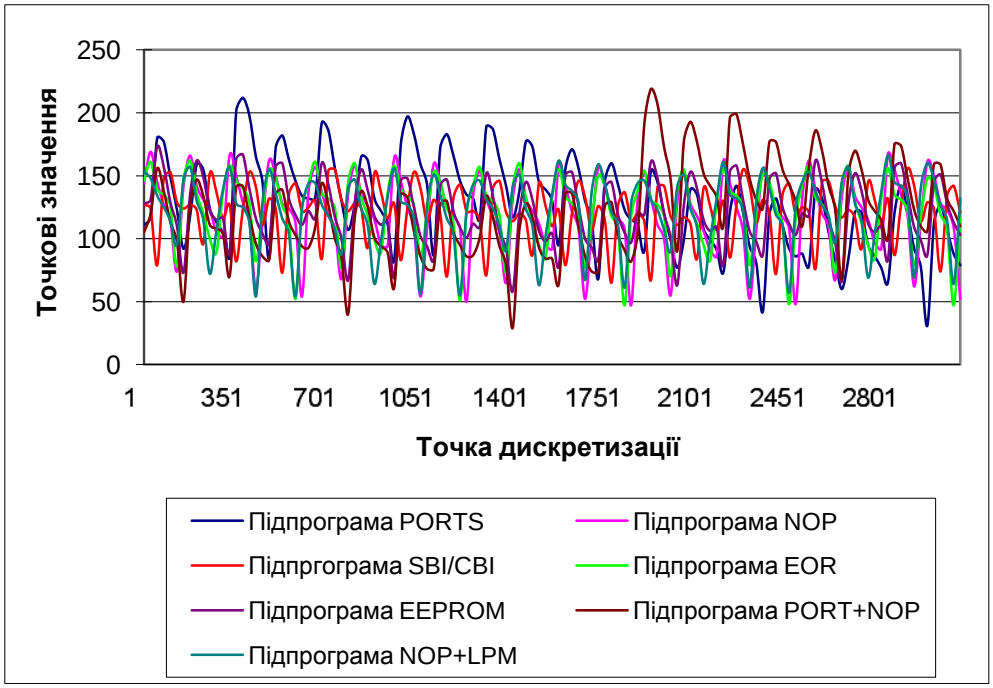


Рис.5.27. Суміщені графіки для аналізу системи захисту на основі маніпуляції внутрішніми ресурсами.

Таблиця 5.10

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT
NOP	0,186909					
SBI/CBI	0,334827	0,494876				
EOR	0,200997	0,465065	0,506081			
EEPROM	0,230023	0,315746	0,43461	0,540149		
NOP+PORT	-0,33374	0,480883	0,444232	0,53977	0,518905	
NOP+LPM	0,079062	0,468305	0,218187	0,361593	0,555693	0,167689

Побудована на основі числових значень табл.5.10 тривимірна гістограма (рис 5.28) показує суттєве збільшення діапазону можливих коефіцієнтів кореляції між струмами споживання при виконанні підпрограм.

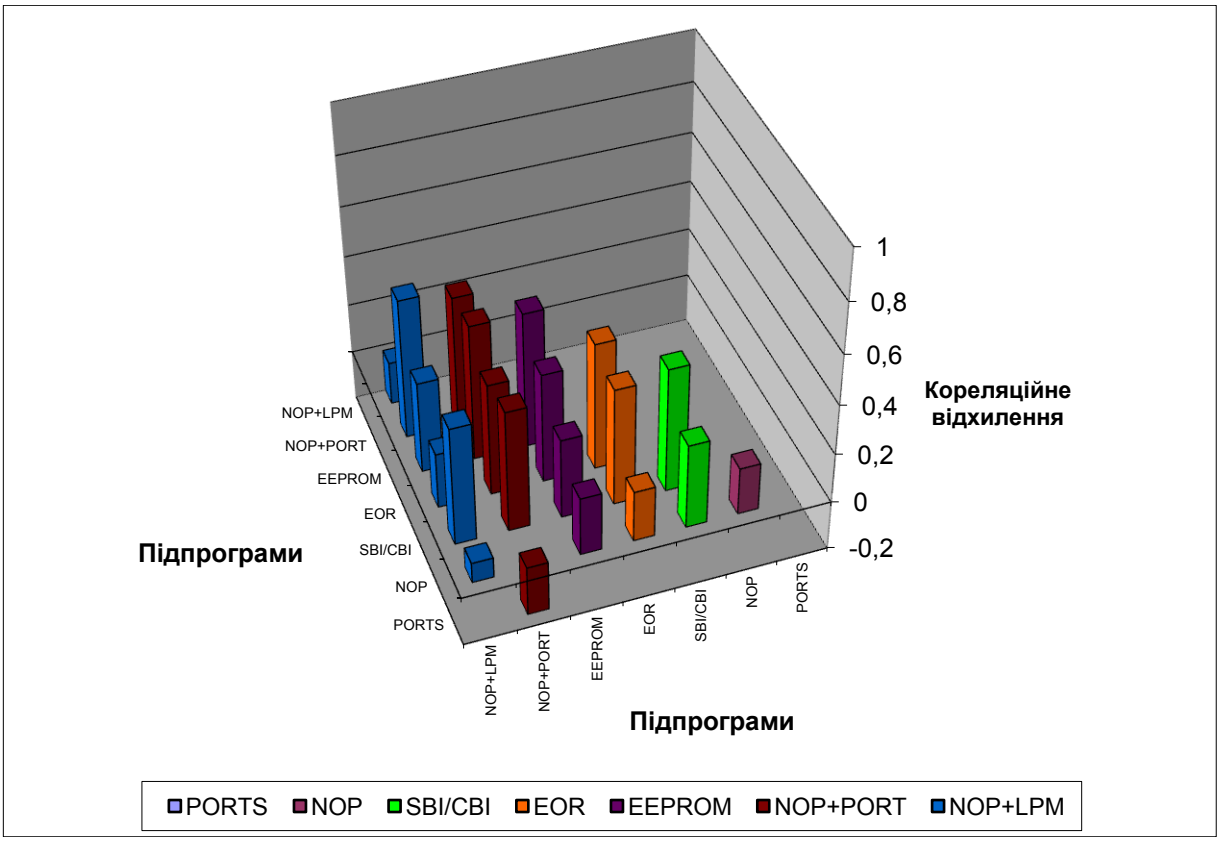


Рис.5.28. Гістограма відхилень для аналізу системи захисту на основі маніпуляції внутрішніми ресурсами.

Розрахунковий коефіцієнт відхилення Δ_{max} становить 0,89, тобто близько 90% ефективного маскування струмів споживання.

Моделювання розробленої *системи захисту від атак за струмом споживання на основі двохядерної структури* дало можливість отримати суміщені графіки семи струмів споживання, що близькі до хаотичного шуму (рис.5.29).

Мінімальні та максимальні значення коефіцієнтів кореляції між струмами виконання підпрограм наведені в табл.5.11 та табл.5.12 відповідно. На їх основі визначена трикутна матриця відхилень, що відображена в табл.5.13 і візуалізована на рис 5.30.

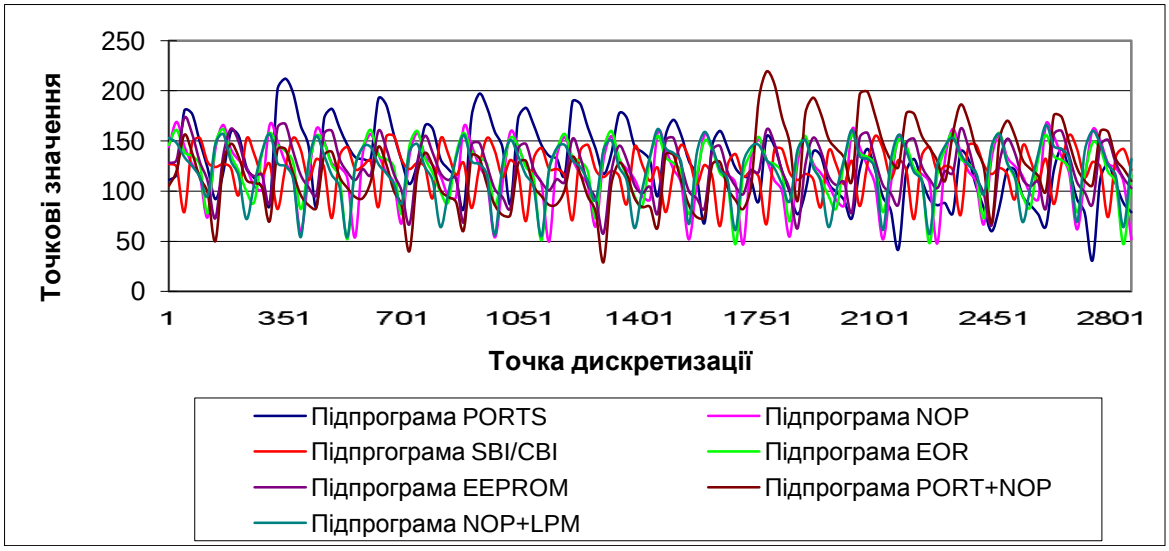


Рис.5.29. Суміщені графіки для аналізу системи захисту на основі двохядерної структури.

Таблиця 5.11

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS	1	-0,36055	-0,28515	-0,18933	-0,31976	-0,43109	-0,3207
NOP	-0,36055	1	-0,2562	-0,21831	-0,20798	-0,10877	-0,12098
SBI/CBI	-0,28515	-0,2562	1	-0,2063	-0,18578	-0,19676	-0,10413
EOR	-0,18933	-0,21831	-0,2063	1	-0,09742	-0,22691	-0,28014
EEPROM	-0,31976	-0,20798	-0,18578	-0,09742	1	-0,08945	-0,12448
NOP+PORT	-0,43109	-0,10877	-0,19676	-0,22691	-0,08945	1	-0,14368
NOP+LPM	-0,3207	-0,12098	-0,10413	-0,28014	-0,12448	-0,14368	1

Таблиця 5.12

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS	1	0,596609	0,492945	0,516035	0,64278	0,211154	0,355222
NOP	0,596609	1	0,73344	0,88581	0,951975	0,766291	0,765794
SBI/CBI	0,492945	0,73344	1	0,644236	0,612701	0,665346	0,941793
EOR	0,516035	0,88581	0,644236	1	0,9119	0,817116	0,854461
EEPROM	0,64278	0,951975	0,612701	0,9119	1	0,816308	0,552856
NOP+PORT	0,211154	0,766291	0,665346	0,817116	0,816308	1	0,5945
NOP+LPM	0,355222	0,765794	0,941793	0,854461	0,552856	0,5945	1

Таблиця 5.13

	PORTS	NOP	SBI/CBI	EOR	EEPROM	NOP+PORT	NOP+LPM
PORTS							
NOP	0,236058						
SBI/CBI	0,207797	0,477243					
EOR	0,326704	0,667501	0,437938				
EEPROM	0,323019	0,743995	0,426916	0,814481			
NOP+PORT	-0,21994	0,657526	0,468587	0,590202	0,726855		
NOP+LPM	0,03452	0,644818	0,837663	0,574317	0,428372	0,450821	

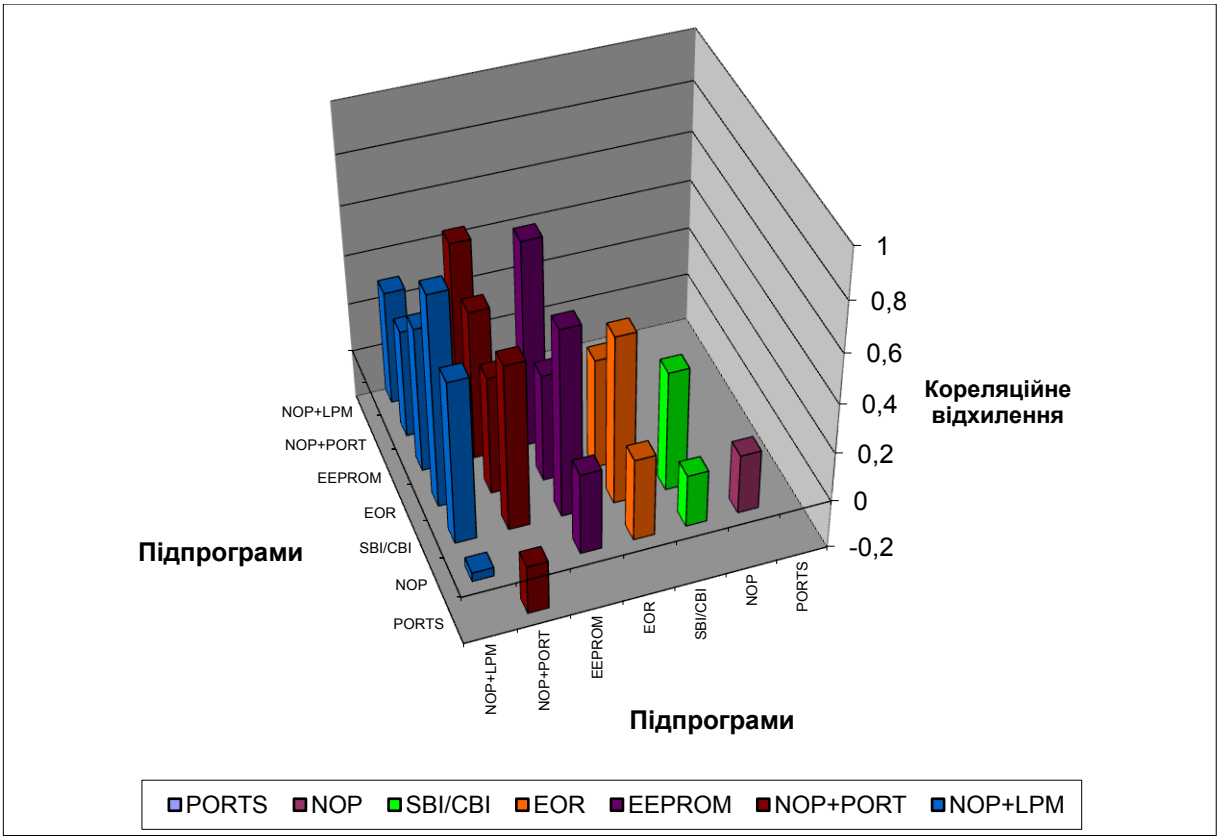


Рис.5.30. Гістограма відхилень для аналізу системи захисту на основі двох ядерної структури.

Результатом дослідження системи захисту на основі двох ядерної структури є визначення її ефективності. Як і у випадку з іншими системами захисту для оцінки ефективності застосовано критерій максимального відхилення коефіцієнтів кореляції та отримаємо числове значення $\Delta_{max} = 1{,}16$, що свідчить про низьку ймовірність детектування в струмі споживання захищеного такою системою мікроконтролера раніше виміряної ділянки динамічного струму споживання. Максимальне зменшення ймовірності детектування досліджуваних підпрограм за табл.5.13 становить 83,7%, середнє значення – 47%.

Якщо прийняти за одиницю виміру максимальне відхилення найменш ефективної системи захисту, то можна побудувати сумарну порівняльну діаграму у відносних одиницях (рис.5.31).

Рис.5.31. Порівняльна діаграма систем захисту від атак за струмом споживання.

Для системи захисту на основі двоядерної структури максимальне відхилення коефіцієнту кореляції становить $\Delta_{max} =$ 1,06. Максимальне зменшення ймовірності детектування підпрограм становить 83,7%, середнє – 47%.

Порівняльний аналіз свідчить, що розроблений регульований фільтр на базі двоядерної структури в 52 рази ефективніший за дві найменш ефективні системи захисту, та в 1,13 рази ефективніший за найближчий з аналогів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Anderson R.J. Low Cost Attacks on Tamper Resistant Devices / R.J. Anderson, Markus G. Kuhn [in M.Lomas et al. (ed.)] // Security Protocols, 5th International Workshop. – Paris, France, April 7-9, 1997.
2. Biham E. Diffrential Fault Analysis of Secret Key Cryptosystems / E. Biham, A. Shamir // Advances in Cryptology: Proceedings of CRYPTO'97. – Springer-Verlag, Au¬gust 1997. – pp. 513-525.
3. Kocher P. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems, / P. Kocher // Advances in Cryptology: Proceedings of CRYPTO'96. – Springer-Verlag, August 1996. – pp. 104-113.
4. Rivest R.L. A method for obtaining digital signatures and public-key cryptosystems / R.L. Rivest, A. Shamir, L.M. Adleman // Communications of the ACM, 21. – 1978. – pp. 120-126.
5. Kocher P. Differential Power Analysis / P. Kocher, J. Jaffe, B. Jun // Crypto 99 Proceedings, Lecture Notes in Computer Science. – M. Wiener, ed., Springer-Verlag, 1999. — Vol.1666.
6. Skorobogatov S.P. Semi-invasive attacks - A new approach to hardware security analysis [Електронний ресурс]. / Sergei P. Skorobogatov – April 2005, 144 p. Режим доступу: http://www.cl.cam.ac.uk/techreports/ UCAM-CL-TR-630.pdf – – Назва з екрану.
7. Atmel 8-bit Microcontroller with 16K Bytes In-System Programmable Flash. ATmega16. [Електронний ресурс]. Режим доступу: http://www.atmel.com/Images/doc2466.pdf – Назва з екрану.
8. Abraham D.G. Transaction Security System / D.G. Abraham, G.M. Dolan, G.P. Double, J.V. Stevens // IBM Systems Journal. – 1991. – Vol. 30(2). – pp. 206–229.
9. TIPS for FIPS 140, Selling Applications with Cryptography to Federal Agencies.

RSA Security White Paper [Електронний ресурс]. Режим доступу: http://wp.bitpipe.com/resource/org_1039183786_34/FIPS_WP_0603_bitpipe.pdf

10. Скоробогатов С.П. Атаки методом оптического наведения ошибок. [Електронний ресурс] / С.П. Скоробогатов, Р.Дж. Андерсон. Режим доступу: http://www.cl.cam.ac.uk/~sps32/optofault_rus.pdf.
11. Atmel 8-bit Microcontroller with 1K Byte Flash ATtiny11, ATtiny12. [Електронний ресурс]. Режим доступу: http://www.atmel.com/Images/1006S.pdf – Назва з екрану.
12. Boneh D. On the Importance of Checking Cryptographic Protocols for Faults / D. Boneh, R. DeMillo, R. Lipton // Advances in Cryptology: Proceedings of EURO-CRYPT'97. – Springer-Verlag, May 1997. – pp. 37-51.
13. Skorobogatov S.P. Using Optical Emission Analysis for Estimating Contribution to Power Analysis [Електронний ресурс] / Sergei Skorobogatov – Computer Laboratory, University of Cambridge, Cambridge, United Kingdom, 2005 Режим доступу: http://www.cl.cam.ac.uk/~sps32/fdtc2009-opt-emis.pdf – Назва з екрану.
14. Пат. WO 02/09030A1, МПК G06K 19/073. Data-processing arrangement comprising confidential data / PAUTOT, Fabrice; filed 11.07.2001, published 31.01.2002.
15. Пат. WO 2004/095366A1, МПК G06K 19/073, Electronic circuit device for cryptographic applications. / Pessolano Francesco; filed 21.04.2004, published 4.11.2004.
16. Пат. US 6264108B1 США, МПК G06K 19/00. Protection of sensitive information contained in integrated circuit cards / Michael Baenstch; filed Jun. 7, 1999, published Jul. 24, 2001.
17. Пат. US 5998978 США, МПК G05F 1/56. Apparatus and method for reducing energy fluctuation in a portable data device / Lawrence Edwin Connel, Patric Lee

Rakers; filed Jun. 29, 1998, published Dec. 7, 1999.

18. Пат. US 6766455B1 США, МПК G06F 11/30, G06F 17/60. System and method for preventing differential power analysis attacks (DPA) on a cryptographic device / Frederick W. Ryan; filed Dec. 9, 1999, published Jul. 20, 2004.
19. Пат. WO 2004/025444A2, МПК G06F 1/26. Current source for cryptographic processor / Hubert Gerardus; filed 29.08.2003, published 25.03.2004.
20. Пат. US 6419159B1 США, МПК G06K 19/06. Integrated Circuit Device With Power Analysis Protection Circuitry / Odinak Gilad; filed Jun. 16, 1999, published Jul. 16, 2002.
21. Пат. US 2002/0024070A1 США, МПК H01L 29/80, H01L 31/112. Integrated circuit with protection device / Richard Fournel; filed Jun. 29, 2001, published Feb. 28, 2002.
22. Пат. US 6748535B1 США, МПК G06F 1/26, G06F 12/14. System and method for suppressing conducted emission by a cryptographic device comprising an integrated circuit / Frederic W. Ryan, Monroe A. Weiant, Edward J. Twarog; filed Dec. 9, 1999, published Jun8, 2004.
23. Пат. US 6848619B1 США, МПК G06K 19/06. Micro-controller protected against current attacks. Robert Leydier; filed Jul. 17, 2000, published Feb. 1, 2005.
24. Пат. US 2002/0010871 США, МПК G06F 1/26; G06F 1/32. Data carrier for the adaptation of a consumption time interval to the power consumption of the data carrier / Peter Thueringer, Klaus Ully, Marcus Feuser; filed May 30, 2001, published Jan 24, 2002.
25. Пат. EP1113386A2, МПК G06K 19/073. Protecting smart cards from power analysis with detached power supplies. / Adi Shamir, Rehovot; filed 23.12.2000, published 04.07.2001.
26. Бойко В.І. Схемотехніка електронних систем: У 3 кн. Кн. 3.

Мікропроцесори та мікроконтролери: Підручник / В.І. Бойко, А.М. Гуржій, В. Я. Жуйков [та ін.] — 2-ге вид., допов. і переробл. — К.: Вища шк., 2004. — 399 с.

27. Жданкин В. Преобразователи напряжения для современных высокопроизводительных цифровых систем / В. Жданкин // Современные технологии автоматизации. – 2002. – №4. – С. 40-50.
28. Zhuikov V. Intellectual systems to control energy generation and consumption in local objects / V. Zhuikov, Yu Petergerya // Proceeding of 2-nd Conference “Power Electronic Devices Compatibility” PEDC. – Poland, Zielona Gora: Technical University Press. – 2001. – pp.208-212.
29. Айфичер Эммануил С. Цифровая обработка сигналов: Практический подход, 2-е издание. [Пер. с англ]. / Эммануил С. Айфичер, Барри У. Джервис – М.: Издательский дом «Вильямс», 2004. – 992 с. : ил.
30. Жуйков В.Я. СКІ-вейвлет-перетворення дискретних функцій / В.Я. Жуйков, Ю.С. Петергеря, Т.А. Хижняк // Технічна електродинаміка. Темат. вип. "Силова електроніка та енергоефективність". – 2003. – Ч.2. – С. 84-87.
31. Жуйков В.Я. Симметричное преобразование на конечных интервалах. / В.Я. Жуйков, Т.А. Терещенко, Ю.С. Петергеря – К.: Аверс, 2000. – 218 с.
32. Петергеря Ю.С. Порівняльний аналіз спектральних перетворень / Ю.С. Петергеря, Ю.В. Хохлов, Т.А. Хижняк // Электроника и связь. – 2002. - №16. – С.71-75.
33. Жуйков В.Я. Спектральные преобразования функций с m-ичным аргументом: теория и применения / В.Я. Жуйков, Т.А. Терещенко, Ю.С. Петергеря – К.: Аверс, 2006. – 293 с.
34. Жуйков В.Я. Преобразование дискретных сигналов на конечных интервалах в ориентированом базисе / В.Я. Жуйков, Т.А. Терещенко, Ю.С. Петергеря – К.: Аверс, 2004. – 274 с.

35. Zhuikov V. Use of discrete transformation at oriented basis for transferring of information during the hindrances / V. Zhuikov, Yu Petergerya // International Workshop on Acoustic Noise and Other Aspects of Power Electronics Compatibility PEDC. - Poland, Slubice: Technical University Press. – 1999. – pp. 87-96.
36. Петергеря Ю.С. Многомерное вейвлет-преобразование в ориентированном базисе / Ю.С. Петергеря , А.А. Гусев // Технічна електродинаміка. Тематичний випуск “Силова електроніка та енергоефективність”. – 2005. – Ч.4. – С.72-75.
37. Жуйков В.Я. Конструирование преобразований дискретных функций на конечных интервалах с заданными свойствами / В.Я. Жуйков, Т.А. Терещенко, Ю.С. Петергеря // Технічна електродинаміка. Тематичний випуск “Проблеми сучасної електротехніки”. – 2004. – Ч.4. – С.7-12.
38. Брондштейн И.Н. Справочник по математике для инженеров и учащихся ВТУЗов. / И.Н. Брондштейн, К.А. Семедяев – М.: Наука, 1980. – 976 с.
39. Анго А. Математика для электро- и радиоинженеров / Андре Анго. – М.: Наука, 1964. – 772 с.
40. Калиткин Н.Н. Численные методы. / Н.Н. Калиткин – М., Наука, 1978. - 512 с.
41. Корн Г.А. Справочник по математике для научных работников и инженеров. / Г.А. Корн, Т.М. Корн – М.: Наука, 1974. – 832 с.
42. Петергеря Ю.С. Быстрые преобразования в ориентированном базисе / Ю.С. Петергеря // Технічна електродинаміка. Тематичний випуск “Силова електроніка та енергоефективність”. – 2004. – Ч.2. – С.123-126.
43. Мороз А.В. Вейвлет-преобразования в полярной системе координат / В.Я. Жуйков, Т.А. Терещенко, Ю.С. Петергеря // Дискретные спектральные преобразования на конечных интервалах / В.Я. Жуйков, Т.А. Терещенко,

Ю.С. Петергеря - К.: НТУУ "КПИ", 2010. – Глава 7. – С.185-201.

44. Мороз А.В. Вейвлет-преобразования дискретных функций в полярной системе координат / А.В. Мороз, Т.А. Терещенко // Электроника и связь. – 2007. – №2. – С.39-46.
45. Зубчук В.И. Справочник по цифровой схемотехнике / В.И. Зубчук, В.П. Сигорский, А.Н. Шкуро – К.: Техника, 1990 – 448 с.
46. Бойко В.І. Схемотехніка електронних систем: У 3 кн. Кн.2. Цифрова схемотехніка: Підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков, А.А. Зорі, [та ін.] — К.: Вища школа, 2004. - 408 с.
47. Гончаров Ю.П. Перетворювальна техніка. Підручник. Частина 2 / Ю. П. Гончаров, О. В. Будьонний, В. Г. Морозов та ін. [За ред. В. С. Руденка]. – Харків: Фоліо, 2000. – 360 с.
48. Руденко В.С. Перетворювальна техніка. Частина 1 / В. С. Руденко, В. Я. Ромашко, В. Г. Морозов. – К.:ІСДО, 1996. – 262 с.
49. Руденко В.С. Преобразовательная техника / В. С. Руденко, В. Н. Сенько, Н. М. Чиженко. – К.: Вища школа, 1983. – 423 с.
50. Boyko V.I. Basics of Circuity of electronics systems: Textbook / V.I. Boyko, V.Y. Zhujkov, A.A. Zori, V.M. Spivak [and others; Translation by E.A.Batina]. - K.: Avers, 2006. – 784 p.
51. Основы промышленной электроники / [под ред. В.Г. Герасимова]. – М.: Высшая школа, 1982.
52. Сташин В.В. Проектирование цифровых устройств на однокристальных микроконтроллерах. / В.В. Сташин, А.В. Урусов, О.Ф. Мологонцева. – М.: Энергоатомиздат, 1990. – 221 с.
53. Бойко В.І. Схемотехніка електронних систем: У 3 кн. Кн.1. Аналогова схемотехніка та імпульсні пристрої: Підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков, А.А. Зорі, [та ін.] — К.: Вища щкола, 2004. — 366 с.

54. Блохин А.В. Теория эксперимента. Курс лекций в двух частях, ч.2 / А.В. Блохин – Мн.: Научно-методический центр “Электронная книга БГУ”, 2003.
55. Єріна А.М. Статистичне моделювання та прогнозування / А. М. Єріна. – К.: КНЕУ, 2001. – 187 с.
56. Давыдов В. Visual C++. Разработка Windows-приложений с помощью MFC и API-функций / Владимир Давыдов. - СПб.: БХВ-Петербург. - 2008. - 576 с.
57. Хортон Айвор Visual C++ 2010. Полный курс / Айвор Хортон. [пер. В. Коваленко]. - М.: Вильямс. - 2011. - 1216 с.
58. Пахомов Б. C/C++ и MS Visual C++ 2010 для начинающих / Б. Пахомов. - СПб.: БХВ-Петербург. - 2011. - 736 с.
59. Шилдт Г. C++: базовый курс, 3-е издание / Герберт Шилдт — М.: "Вильямс", 2012. — 624 с.
60. Ануфриев И.Е. Самоучитель MatLab 5.3/6.x. / И.Е. Ануфриев – СПб.: БХВ-Петербург, 2004. – 736 с.: ил.
61. Гандер В. Решение задач в научных вычислениях с применением Maple и MATLAB / В. Гандер - М.: Вассамедина, 2005. – 311 с.
62. Дьяконов В.П. MATLAB 6/6.1/6.5 + Simulink 4/5 в математике и моделировании / В.П.Дьяконов. - М.: СОЛОН-Пресс, 2003.
63. Кетков Ю. Matlab 7: Программирование, численные методы / Ю. Кетков, А. Кетков, М. М. Шульц. – СПб. : БХВ-Петербург, 2005 . - 737 с.
64. Бойко В.І. Основи схемотехніки електронних систем: Підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков, А.А. Зорі [та ін.] — К.: Вища щкола, 2004. — 536 с.
65. Якименко Ю.І. Мікропроцесорна техніка: Підручник / Ю.І. Якименко, Т.О Терещенко, Є.І. Сокол, В.Я. Жуйков, [та ін.; За ред. Т.О. Терещенко]. - К.:

Видавництво "Політехнік", 2003.- 440 с.

66. Иглин С.П. Теория вероятностей и математическая статистика на базе MATLAB. / С.П. Иглин. - Харьков: НТУ "ХПИ" - 2006. - 612 с.
67. Курбатова Е.А. MATLAB 7. Самоучитель. / Е.А. Курбатова - М.: Вильямс. - 2005. - 256 с.
68. Худяков В.Ф. Моделирование источников вторичного электропитания в среде MATLAB 7.x: учебное пособие. / В.Ф. Худяков, В.А. Хабузов. - СПб.: ГУАП, 2008, 332 с.
69. Черных И.В. Моделирование электротехничеких устройств в MATLAB, SimPowerSystems и Simulink. / И.В. Черных. - М.: ИД Питер. - 2007. - 288 с.
70. Дэбни Дж. Simulink 4. Секреты мастерства. / Дж. Дэбни, Т. Харман [пер. М. Симонова]. - М.: Бином. Лаборатория знаний. - 2003. - 404 с.
71. Пат. GB2398139A Великобританія, МПК G06K 19/073. Smart cards having protection circuits therein that inhibit power analysis attacks. / Seo-Kyu Kim; filed 23.01.2004, published 11.08.2004.
72. Пат. US005965912A США, МПК H01L 27/108. Variable Capacitor and Method for Fabricating the same. / David Lewis Stolfa, Kenneth D. Cornett; filed Sep. 3, 1997, published Oct. 12, 1999.
73. Application Note 3469. Building a Low-Cost White-Noise Generator [Електронний ресурс] / Maxim Integrated, 2005. Режим доступу: http://www.maximintegrated.com/app-notes/index.mvp/id/3469 – Назва з екрану.
74. Лебедев О.Н. Микросхемы памяти и их применение / О.Н. Лебедев. - М.: Радио и связь, 1990 – 234 с.
75. Корнеев В.В. Современные микропроцессоры. / В.В. Корнеев А.В. Киселев. - М.: НОЛИДЖ, 1998. - 240 с.
76. IBM Cryptographic Products. IBM PCI Cryptographic Coprocessor. General

Information Manual. [Електронний ресурс] / IBM, 2002. Режим доступу: ftp://www6.software.ibm.com/software/cryptocards/IBM%204758%20Gen%20Info%202002-05-30.pdf – Назва з екрану.

77. Терещенко Т.А. Математические основы прогнозного управления полупроводниковыми преобразователями / Т.А. Терещенко , Ю.С. Петергеря , Н.В. Колотов // Технічна електродинаміка. Тематичний випуск “Силова електроніка та енергоефективність”. – 2006. – Ч.3. – С.67-70.
78. Евстифеев А.В. Микроконтроллеры AVR семейства Tiny и Mega фирмы “Atmel” / А.В. Евстифеев – М.: Издательский дом «Додэка-XXI», 2004 – 560 с.
79. Кривченко И.В. Микроконтроллеры общего назначения для встраиваемых приложений производства Atmel Corp. / И.В. Кривченко // Электронные компоненты, №5, 2002. - с. 69-73.
80. Баранов В.Н Применение микроконтроллеров AVR: схемы, алгоритмы, программы. / В.Н Баранов. - М.: Издательский дом "Додэка XXI", 2001. - 288 с.
81. Шилдт Г. C: полное руководство, классическое издание / Герберт Шилдт — М.: "Вильямс", 2011. — 704 с.
82. Шилдт Г. Справочник программиста по C/C++, 3-е издание / Герберт Шилдт. — М.: "Вильямс", 2006. — 432 с.
83. Астафьева Н.М. Вейвлет-анализ: основы теории и примеры применения / Н. М. Астафьефа // Успехи физических наук. – 1996. – №166 (11). – С.1145-1170.
84. Воробьев В.И. Теория и практика вейвлет–преобразования / В.И. Воробьев, В.Г. Грибунин – СПб.: Военный университет связи, 1999. – 208 с.
85. Яковлев А.Н. Основы вейвлет-преобразования сигналов / А. Н. Яковлев. – М.: Сайнс пресс, 2003. – 176 с.

86. Капшій О. Вейвлет-перетворення у компресії та попередній обробці зображень / О. Капшій, О. Коваль, Б. Русин. — Львів: СПОЛОМ, 2008. — 208 с.
87. Дремин И.М. Вейвлеты и их использование / И.М. Дремин, О.В. Иванов, В.А. Нечитайло // Успехи физических наук. – 2001. – №171(5). – С.465-501.
88. Геранін В.О. Теорія вейвлетів з елементами фрактального аналізу: Науково-методичне видання. / В.О. Геранін, Л.Д. Писаренко, Я.Я. Рущицький – К.: ВПФ УкрІНТЕІ, 2002. – 364 с.
89. Сокол Є.І. Розробка наукових основ створення сучасних напівпровідникових перетворювачів електроенергії та їх впровадження в системах живлення статичних та динамічних навантажень / Є. І. Сокол, В. Б. Клєпіков, К. О. Липківський [та ін.] // Технічна електродинаміка. Тематичний випуск «Проблеми сучасної електротехніки». – 2000. – Ч.1 – С.46-49.
90. Мороз А.В. Система керування джерелом живлення з інформаційною шиною для захисту даних за струмом споживання / А.В. Мороз // Технічна електродинаміка. Тем. випуск "Силова електроніка та енергоефективність". – 2010.
91. Мороз А.В. Обробка даних датчика положення мікросупутника в полярних координатах. / А.В. Мороз, Т.А. Терещенко, І.В. Хохлов // Технічна електродинаміка. Тем. випуск "Силова електроніка та енергоефективність". – 2008. – ч.3. – С.100-102.
92. Мороз А.В. Дослідження захищеності програмного забезпечення мікроконтролерів за струмом споживання / А.В. Мороз, Т.О. Терещенко // Технічна електродинаміка. Тем. випуск "Проблеми сучасної електротехніки". – 2008. – ч.2.– С.99-102.
93. Мороз А.В. Визначення рівня захищеності програмного забезпечення

мікроконтролерів за методом аналізу струму споживання / А.В. Мороз // Электроника и связь. Тем. выпуск "Проблемы электроники". – 2008. – ч.1. – №1-2. – С.238-241.

94. Беженар В.О. Цифрова система захисту від атак за струмом споживання, / В.О. Беженар, А.В. Мороз, Т.О. Терещенко // Электроника и связь. – 2010. – №2. – С.108-114.
95. Беженар В.О. Захист інформації від зчитування за струмом споживання з використанням генератора випадкових чисел. / В.О. Беженар, А.В. Мороз, Т.О. Терещенко // Технічна електродинаміка. Тем. випуск "Силова електроніка та енергоефективність". – 2009. – ч.1. – С.39-42.
96. Пат. UA 43673 Україна, МПК G06K 19/06 (2006.01). Мікроконтролер з системою захисту від атак за струмом споживання. / В.О. Беженар, А.В. Мороз, Т.О. Терещенко; заявл. 03.04.2009 № u200903207, опубл. 25.08.2009.
97. Пат. UA 43634 Україна, МПК G06F 1/00 (2006.01). Мікропроцесорна система з захистом від зчитування за струмом споживання / В.О. Беженар, А.В. Мороз, Т.О. Терещенко; заявл. 25.03.2009 №u200902780, опубл. 25.08.2009.
98. Електронний підручник «Енергетична електроніка» [Електронний ресурс] / В.Я. Жуйков, В.В. Рогаль, О.В. Будьоний, В.В. Пілінський [та ін.]. – К., 2008. – Режим доступу: http://www.kaf-pe.kpi.ua/books/EE.zip – Назва з екрану.
99. Електронний підручник "Мікропроцесори та мікроконтролери" [Електронний ресурс] / В.Я. Жуйков, Т.А. Терещенко, Ю.С.Петергеря, Ю.В. Хохлов, А.В. Мороз – К., 2009. – Режим доступу: http://www.kaf-pe.kpi.ua/books/MPT.zip – Назва з екрану.
100. Сокол Е.И. Принципы построения микропроцессорных систем управления полупроводниковыми преобразователями / Ю.И. Якименко, В.Я. Жуйков,

М.Р. Вержановская // Технічна електродинаміка. Тематичний випуск „Силова електроніка та енергоефективність”. – 2001. – Ч.3. – С. 43-45.

101. Шидловский А.К. Полупроводниковые преобразователи электротранспортных средств специального назначения / В.Б. Павлов, О.Н. Юрченко, А.В. Попов, В.Е. Павленко // Технічна електродинаміка. Тематичний випуск „Силова електроніка та енергоефективність”. – 2004. – Ч. 2. – С.24-25.

102. Кириленко О.В. Інтелектуальні системи керування потоками електроенергії у локальних об’єктах / О.В. Кириленко, Ю.С. Петергеря, Т.О. Терещенко, В.Я. Жуйков – К.: Медіа ПРЕС, 2005. – 212 с.

103. Мороз А.В. Регульовані фільтри джерел живлення мікроконтролерів із захистом інформації / А.В. Мороз // Електроніка-2011 - IV міжнародна науково-технічна конференція молодих вчених - Збірник статей, 2011. – Частина 1 – С.17-21.

104. Ямненко Ю.С. Технічний захист конфіденційної інформації в мікроконтролерах. / Ю.С.Ямненко, А.В.Мороз // Спеціальна техніка у правоохоронній діяльності – Матеріали V Міжнародної науково-практичної конференції (Україна, Київ, 25 листопада 2011 року), 2011. – С.255-25.

105. Терещенко Т.О. Теореми спектрального аналізу перетворення в орієнтованому базисі / Т.О. Терещенко, Ю.С. Петергеря, Ю.В. Хохлов // Электроника и связь. – 2001. – №13. – С. 24-28.

106. Юрченко О.М. Високочастотні транзисторні перетворювачі у системах електроживлення технологічних установок / О.М. Юрченко, М.М. Юрченко, В.Я. Гуцалюк, В.О. Павловський [та ін.] // Праці ІЕД НАНУ [Збірних наукових праць]. - 2009. - Випуск 23. - С. 118-127.

ДОДАТОК А Вейвлет-аналіз дискретних функцій

Вейвлет-перетворення Хаара

Найбільш поширеним і простим для розрахунків є вейвлет-аналіз на базі функцій Хаара [83] [84] [85] [86]. Вхідний сигнал $f(x)$ з кількістю відліків 2^{j_m} для вейвлет-перетворення Хаара описується у вигляді:

$$f(x) = \sum_{k=0}^{2^{j_m}} s_{j_m,k} \varphi_{j_m,k}(x) \qquad \text{(A.1)}$$

де $s_{j_m,k} = f(k/2^{j_m})/2^{\frac{j_m}{2}}$, j_m- максимальний рівень розкладання функції, $\varphi_{j_m,k} = 2^{j_m/2}\varphi\left(2^{j_m}x - k\right)$.

Базисними функціями перетворення Хаара [87] [83] [88] є скейлінг-функція $\varphi(x) = \{1;1\}$ і "материнский вейвлет" $\psi(x) = \{1;-1\}$ (рис. А.1).

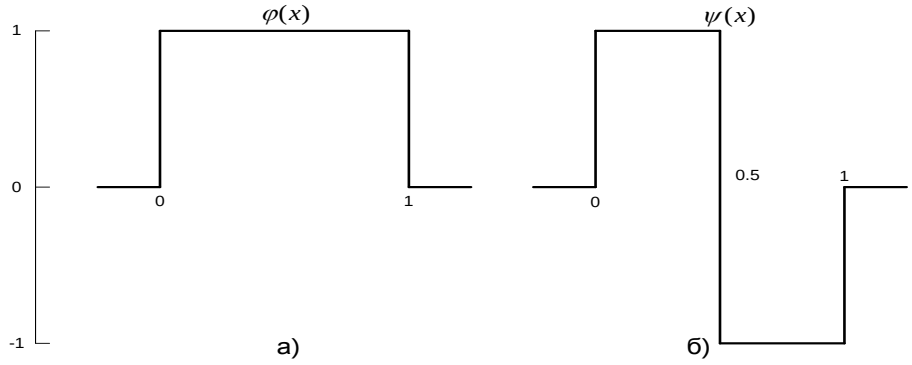


Рис. А.1. Скейлінг-функція $\varphi(x)$ (а) та „материнський вейвлет” $\psi(x)$ (б)

Вейвлет-перетворення являє собою ітераційну процедуру при якій аналіз функції проводиться на різних рівнях розкладання з поступовим розширенням інтервалу розгляду від мінімального, котрий включає 1/2 відліку, до максимального, рівного інтервалові визначення функції.

Аналітичні співвідношення для нормувальних сум і різниць на j рівні розкладання мають вигляд:

$$\begin{aligned} s_{j-1,k} &= \frac{1}{\sqrt{2}}\left[s_{j,2k} + s_{j,2k+1}\right] \\ d_{j-1,k} &= \frac{1}{\sqrt{2}}\left[s_{j,2k} - s_{j,2k+1}\right] \end{aligned} \qquad \text{(A.2)}$$

Зворотне перетворення, тобто відновлення функції-оригіналу, виконується по формулах:

$$\begin{aligned} s_{j,2k} &= \frac{1}{\sqrt{2}}\left[s_{j-1,k} + d_{j-1,k}\right] \\ s_{j,2k+1} &= \frac{1}{\sqrt{2}}\left[s_{j-1,k} - d_{j-1,k}\right] \end{aligned} \qquad \text{(A.3)}$$

Процес усереднення функції-оригіналу виконується до одержання середнього значення по всьому інтервалу N, що позначається як $s_{0,0}$. Найбільший інтервал розгляду дорівнює інтервалу визначення функції-оригіналу, а найменший – одному відліку. Відповідна нормована різниця на цьому інтервалі позначається як $d_{0,0}$, а функція-оригінал записується як:

$$f(x) = s_{0,0}\varphi_{0,0}(x) + d_{0,0}\psi_{0,0}(x) + \sum_{k=0}^{1} d_{1,k}\psi_{1,k}(x) + \ldots + \sum_{k=0}^{2^j-1} d_{j-1,k}\psi_{j-1,k}(x) \qquad \text{(A.4)}$$

У загальному випадку кожна функція-оригінал записується через коефіцієнти усереднення $s_{j,k}$ і деталізації $d_{j,k}$, розраховані для даного значення $j = j_n$:

$$f(x) = \sum_{k=0} s_{j_n,k}\varphi_{j_n,k}(x) + \sum_{k=0} d_{j_n,k}\psi_{j_n,k}(x) \qquad \text{(A.5)}$$

Таке представлення функції-оригіналу містить інформацію про флуктуації, що є у функції-оригіналі, на обраному рівні розгляду j_n.

Інформація про величину флуктуацій функції-оригіналу представляється коефіцієнтами $d_{j,k}$ при дії на функцію „вузькополосного” фільтра з імпульсною характеристикою $\psi_{0,0}(x)$, і розглядається як високочастотна інформація про поведінку функції-оригіналу. Дія на функцію „широкополосного” фільтра з імпульсною характеристикою $\varphi_{0,0}(x)$ дозволяє одержати інформацію про середні значення функції на деяких інтервалах, тобто низькочастотну інформацію у виді коефіцієнтів $s_{j,k}$.

Таким чином, вейвлет-перетворення на базі функцій Хаара дозволяє виділяти умовно низькочастотну і високочастотну інформацію про сигнал[8], використовуючи тільки два фільтри. Це дозволяє спростити процес розрахунків, що є істотною перевагою даного підходу, але одночасно обмежує обсяг інформації, одержуваної про флуктуації функції-оригіналу, оскільки використовується тільки один високочастотний фільтр. Крім того, використання як вейвлет-функцій базисних функцій Хаара вводить обмеження на кількість дискретних відліків функції-оригіналу, що повинна бути кратна двом.

Вейвлет-перетворення на базі функцій ОБ-перетворення

В якості базисних функцій вейвлет-перетворення використовують базисні функції ОБ-перетворення [33] на інтервалі $N = 3$.

Базисні функції прямого та зворотного ОБ-перетворення відповідно визначаються як:

$$\varphi_d(\nu, x) = \cos\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right] + tg\alpha\sin\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right] \qquad \text{(A.6)}$$

$$\varphi_r(\nu, x) = \cos\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right] + ctg\alpha\sin\left[\frac{2\pi}{m}\sum_{s=1}^{n}\nu^{(s)}x^{(s)}\right] \qquad \text{(A.7)}$$

де $\alpha=2\pi i/m$, $i=1...(m-1)$ – кут орієнтації вісі перетворення;

$x^{(s)}$, $\nu^{(s)}$ – розрядні компоненти у m-ічному представленні чисел x та ν,

$\nu=0...(m^n-1)$, $x=0...(m^n-1)$, n – ціле додатне число.

Базисна функція прямого ОБ-перетворення $\varphi_d(\nu,x)$ для значень $\nu=0$, $x=0...(m-1)$ має вигляд одиничної функції, саме тому її використовують у якості скейлінг-функції ОБ вейвлет-перетворення.

Базисні функції, визначені для $\nu=1...(m-1)$, приймають в якості "материнські вейвлети". Позначимо материнські вейвлети на $N = 3^n$ для $\nu=1$ та $\nu=2$ відповідно як $\psi_d(x)$ та $\gamma_d(x)$.

Запишемо функції для прямого та зворотного перетворення у матричному вигляді (матрицю базисних функцій прямого вейвлет-перетворення позначимо як D і зворотного вейвлет-перетворення як I):

$$D = \begin{vmatrix} \varphi_d \\ \psi_d \\ \gamma_d \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{vmatrix} \tag{A.8}$$

$$I = \begin{vmatrix} \varphi_r \\ \psi_r \\ \gamma_r \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{vmatrix} \tag{A.9}$$

Вид функцій $\varphi_d(x) = \{1;1;1\}$, $\psi_d(x) = \{1;-1;0\}$ і $\gamma_d(x) = \{1;0;-1\}$ показано на рис.А.2.

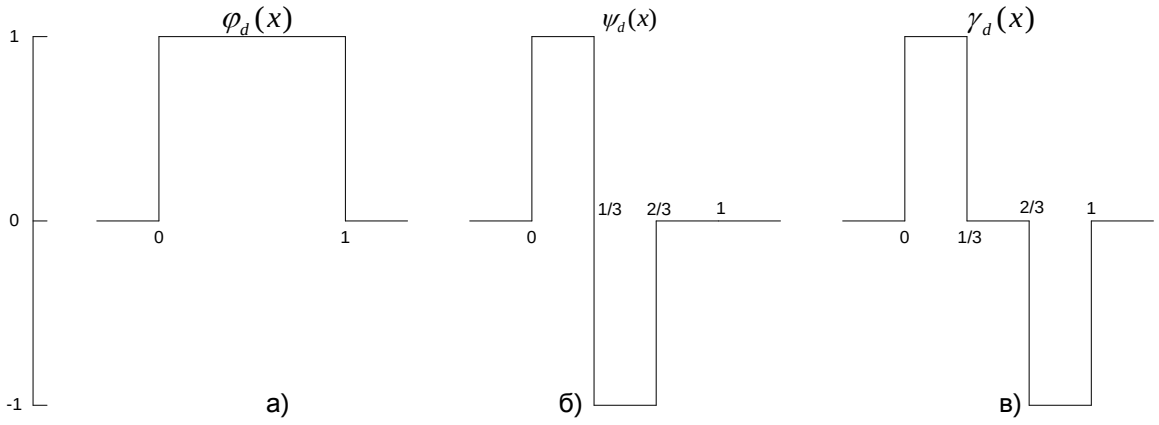


Рис. А.2. Скейлінг-функція $\varphi_d(x)$ (а) та "материнські вейвлети" $\psi_d(x)$ (б) і $\gamma_d(x)$ (в)

Базис вейвлет-функцій з використанням функцій прямого і зворотного ОБ-перетворення $\varphi(x)$, $\psi(x)$, $\gamma(x)$ формується в такий спосіб:

$$\begin{aligned} \varphi_{j,k} &= 3^{j/2}\varphi\left(3^j x - k\right) \\ \psi_{j,k} &= 3^{j/2}\psi\left(3^j x - k\right) \\ \gamma_{j,k} &= 3^{j/2}\gamma\left(3^j x - k\right) \end{aligned} \tag{A.10}$$

де φ приймає значення φ_d або φ_r, ψ приймає значення ψ_d або ψ_r а γ приймає значення γ_d або γ_r в залежності від напрямку вейвлет-перетворення (пряме або зворотне).

Система базисних функцій ОБ-перетворення має властивості ортогональності на інтервалі визначення N і повноти, тобто виконується співвідношення:

$$D \cdot I = \begin{vmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{vmatrix} \cdot \begin{vmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{vmatrix} = 3 \cdot \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} \tag{A.11}$$

Розглянемо функцію-оригінал $f(x)$, яка визначається кількістю відліків 3^{j_m} .

$$f(x) = \sum_{k=0}^{3^{j_m}} s_{j_m,k} \varphi_{j_m,k}(x) \tag{A.12}$$

де

$$s_{j_m,k} = \frac{f(k/3^{j_m})}{3^{\frac{j_m}{2}}} \tag{A.13}$$

і $\varphi_{j_m,k}$ визначається як сходинка з одиничною нормою і шириною $1/3^N$, відмінна від нуля тільки на k-му відрізку.

Оскільки використовуються три різні фільтри, то розкладання функції-оригіналу буде виконуватися по трьом видам коефіцієнтів, що розраховуються наступним чином:

$$\begin{aligned} s_{j-1,k} &= \frac{1}{\sqrt{3}}\left[s_{j,3k} + s_{j,3k+1} + s_{j,3k+2}\right], \\ d_{j-1,k} &= \frac{1}{\sqrt{3}}\left[s_{j,3k} - s_{j,3k+1}\right], \\ l_{j-1,k} &= \frac{1}{\sqrt{3}}\left[s_{j,3k} - s_{j,3k+2}\right] \end{aligned} \tag{A.14}$$

Співвідношення для зворотного перетворення визначається через коефіцієнти розкладання (А.14) як:

$$
\begin{aligned}
s_{j,3k} &= \frac{1}{\sqrt{3}}\left[s_{j-1,k} + d_{j-1,k} + l_{j-1,k}\right], \\
s_{j,3k+1} &= \frac{1}{\sqrt{3}}\left[s_{j-1,k} - 2d_{j-1,k} + l_{j-1,k}\right], \\
s_{j,3k+2} &= \frac{1}{\sqrt{3}}\left[s_{j-1,k} + d_{j-1,k} - 2d_{j-1,k}\right]
\end{aligned}
\qquad \text{(A.15)}
$$

У такий спосіб розрахунок являє собою ітераційну процедуру при якій аналіз функції проводиться на різних рівнях розкладання з поступовим разширенням інтервалу розгляду від мінімального, котрий включає три відліки, до максимального, рівного інтервалу визначення функції.

Виконавши перший крок вейвлет-перетворення запишемо функцію-оригінал у вигляді:

$$
f(x) = \sum_{k=0}^{3^{j_m-1}} s_{N-1,k}\varphi_{N-1,k}(x) + \sum_{k=0}^{3^{j_m-1}} d_{N-1,k}\psi_{N-1,k}(x) + \sum_{k=0}^{3^{j_m-1}} l_{N-1,k}\gamma_{N-1,k}(x) \qquad \text{(A.16)}
$$

де $\psi = \psi_r$, $\gamma = \gamma_r$.

Проводячи обчислення по алгоритму в інтервалах по всіх рівнях j, одержуємо коефіцієнти розкладання.

На інтервалі з найбільшою шириною визначення маємо тільки одне середнє значення по всьому інтервалу, що позначається як $s_{0,0}$. Кожен сигнал можна охарактеризувати його середніми (по деяких інтервалах) значеннями (тренді) і його змінами навколо тренда. Ці коливання навколо усереднених значень називаються флуктуаціями незалежно від причини їхньої появи, будь вони викликані динамічними, стохастичними, психологічними, фізіологічними або якимись іншими факторами. При обробці сигналу звичайно цікавляться величиною флуктуації на різних масштабах, тому що по них одержують відомості про походження цих флуктуацій. Наступний запис функції $f(x)$ містить як середнє значення $s_{0,0}$, так і всі коливання навколо середнього значення:

$$f(x) = s_{0,0}\varphi_{0,0}(x) + d_{0,0}\psi_{0,0}(x) + l_{0,0}\gamma_{0,0}(x) + \sum_{k=0}^{2} d_{1,k}\psi_{1,k}(x) + \sum_{k=0}^{2} l_{1,k}\gamma_{1,k}(x) +$$
$$+ \sum_{k=0}^{8} d_{2,k}\psi_{2,k}(x) + \sum_{k=0}^{8} l_{2,k}\gamma_{2,k}(x) + \ldots + \sum_{k=0}^{N-1} d_{j_m,k}\psi_{j_m,k}(x) + \sum_{k=0}^{N-1} l_{j_m,k}\gamma_{j_m,k}(x) \qquad \text{(A.17)}$$

де $\psi = \psi_r$, $\gamma = \gamma_r$.

Функції $\varphi_{j,k}(x)$, $\psi_{j,k}(x)$ і $\gamma_{j,k}(x)$ описують кінцеву імпульсну характеристику „широкополосного” фільтру і двох „вузькополосних” фільтрів відповідно, тому що вони відфільтровують компоненти сигналу на великих і малих масштабах. Доданки з коефіцієнтами $d_{j,k}$ і $l_{j,k}$, де $j > 0$, в рівнянні (А.24) вказують на флуктуації у все більш дрібних інтервалах з великими j. У загальному випадку усього мається 3^j коефіцієнтів $s_{j,k}$ і по $3^{j_n} - 3^j$ коефіцієнтів $d_{j,k}$ і $l_{j,k}$, де j_m позначає початковий рівень з найменшими інтервалами.

Наведені записи функції-оригіналу $f(x)$ у виразах (А.12), (А.16) та (А.17) еквівалентні математично. Однак останній з них (А.17), що представляє результат вейвлет-перетворення досліджуваної функції, розкриває флуктуаційну структуру сигналу на різних масштабах j і в різних точках k, що утримується в наборі коефіцієнтів $d_{j,k}$ і $l_{j,k}$, тоді як початкова гістограма (А.13) не показує флуктуаційну картину під великим фоном загального тренда. Остаточна формула (А.17) містить загальну середню величину сигналу по всьому інтервалу, представлену коефіцієнтом $s_{0,0}$, і всі його флуктуації з чітко зазначеним масштабом і положенням у кожному з k нормованих коефіцієнтів $d_{j,k}$ і $l_{j,k}$, а початкова гістограма вказує тільки на нормовані середні величини $s_{j,k}$ в k інтервалах. Більш того, при практичному використанні останнє вейвлет-представлення переважніше, тому що для досить гладких функцій, які міняються лише при деяких дискретних значеннях їхніх аргументів, більшість з d і l коефіцієнтів у формулі (А.17) при великому j виявляються дуже маленькими в порівнянні з "інформативними" d і l коефіцієнтами при малому

j і ними можна знехтувати, що спрощує розрахунки і збереження інформації. Смуги нулів (або близьких до нуля значень коефіцієнтів) указують ті області , де функція досить гладка.

Таким чином, компоненти $s_{j,k}\varphi_{j,k}$ відповідають середнім значення (тренди) функції-оригіналу на різних рівнях розкладання j, а $d_{j,k}\psi_{j,k}$ та $l_{j,k}\gamma_{j,k}$ - флуктуаціям, що виділені фільтрами $\psi_r(x)$ та $\gamma_r(x)$, на різних рівнях розкладання j.

З наведених способів аналізу струму споживання ОБ-перетворення та побудоване на його базі ОБ вейвлет-перетворення при m=3 мають найменшу трудомісткість обчислення, а тому мають просту алгоритмічну реалізацію на мікропроцесорах.